No: Y80-102
Title: Software Management Instruction
Rev. Date: 07/19/04
This instruction implements the management requirements for Y80-101PD, "Software Management Program Description" which is the software management program established for the Y-12 National Security Complex, hereafter called Y-12. It establishes and documents activities to be performed, the roles and responsibilities, and the documentation and records required as evidence of performance.

REQUIRED FOR WORKING COPIES ONLY

A working copy of this document is valid only until the document revision number has been changed on the Web. The paper copy should be dated and signed the day it is printed. If you continue to working from the paper copy, you should reverify its accuracy on the Web after seven calendar days.

Date Verified/Reverified	Signature/Initials

BWXT Y-12, L.L.C.

Management Requirements

BWXT Y-12

Procedure

Number: Y80-102

Rev. Date: 7/19/04

Supersedes: 09/30/03

Subject: Software Management Instruction

Debra J. Williams /s/ — 7/19/04

Written by/Signature/Printed Name — Date

Approvals: **[Approval Signatures and Dates on File]**

C. H. Moseley, Jr. /s/ — 7/23/04

Functional Area Manager/Signature/Printed Name — Date

Larry E. Cumberland /s/ — 7/23/04

Executive Manager/Signature/Printed Name — Date

8/09/04

Effective Date

Concurrence:

This document has completed the management requirements process.

S. G. Brown /s/ 8/3/04

Requirements Management

This document has been reviewed by an Authorized Derivative Classifier and UCNI Reviewing Official and has been determined to be unclassified and contains no UCNI. This review does not constitute clearance for public release.

M. W. Sherrill /s/ 7/20/04

Signature and Date

TABLE OF CONTENTS

(Page 1 of 1)

TABLE OF CONTENTS

(Page 1 of 1)

List of Tables

REVISION LOG

Revision Date	Description of Change	Pages Affected
July 19, 2004	DM/R 04-QAO-01, Revise content for clarity and consistency to reflect changes in SQA and to streamline the SQA process.	Added Sections 1.2.6.9 Operating Systems, 1.2.6.10 Common Desktop Productivity Tools, and 1.2.6.11 Limited Use. Sections 1, 2, 3, 4, and Appendices A and B. Previous content in Sections 2 and 3 were merged into the appropriate lifecycle stage in Section 2. All of the Required Deliverables tables were revised to make them easier to follow by removing extraneous information. Deliverables were moved from the Appendices to the appropriate stage in Section 2. Roles and Responsibilities in Section 2 were clarified and re-formatted.
September 30, 2003	DM/R 02-QAO-04, Revision to change the document number and to clarify roles, responsibilities and actions	1.2.2.4, Training and Qualification of Personnel
February 23, 2001	DM/R 01-QAO-01, Revision 1 to clarify selected sections and to make consistent with the Software Application Manager (SAM) which electronically implements this instruction.	Various page sections that were revised are identified by revision bars. Extensive renumbering of sections was necessary due to the deletion and addition of subsections. All of the Required Deliverables and Approvals Matrix tables were revised to make them easier to follow by removing extraneous information.
December 15, 2000	DM/R 00-QAO-16. Total Revision. Number change to Instruction	All
11/01/2000	BWXT Y-12 Blue Sheet to adopt procedure with listed changes.	All

1 Introduction

1.1 Purpose

This instruction implements the management requirements for Y80-101PD, "Software Management Program Description" which is the software management program established for the Y-12 National Security Complex, hereafter called Y-12. It establishes and documents activities to be performed, the roles and responsibilities, and the documentation and records required as evidence of performance. This implementing instruction applies to the following:

- software developed, procured including commercial off-the-shelf applications (COTS), or otherwise obtained from an external or government source which is implemented directly or modified for use and controlled and/or maintained for use at Y-12;
- software internally developed and maintained at Y-12;
- data management (i.e., data generated, collected, stored, managed and used by software); and
- software developed, procured, obtained, or modified for use external to Y-12 in the absence of an accepted alternative Software Quality Assurance (SQA) Program.

This implementing instruction does not apply to software that is developed, procured, obtained, or modified by BWXT Y-12 for a contracting entity where the contracting entity requires the use of an alternative SQA Program. Commercially supplied electronic catalogs such a vendor part specifications and commercially supplied electronic reference sources such as dictionaries, encyclopedias, standards and reference tables are not considered software under this procedure.

1.2 Strategy

This instruction presents an integrated approach for implementing requirements associated with a Software Engineering Methodology, Quality Assurance, and Security. DOE G 200.1-1A, "Software Engineering Methodology (SEM)," is the Software Engineering Standard used for this program. SEM deliverables were customized to meet the SQA needs at Y-12. Exemptions to required deliverables are allowed only with a justification and a rationale on file. Specific exemption instructions are covered in the life cycle stages in Section 2.

The instruction is divided into sections corresponding to specific life cycle stages of software development. The life cycle stages are Planning, Requirements Definition, Functional Design, System Design, Programming, Integration and Testing, Installation and Acceptance, Maintenance, and Retirement.

A key function of the Project Team is to determine the impact of a software failure. A Software Failure Impact (SFI) of 0, 1, 2, or 3 is then assigned. The potential for exemption of a deliverable is also coupled to the source of the software. The Software Project Plan, developed and maintained throughout the life cycle stages, also integrates the Quality Assurance Plan for the software project.

1.2.1 Instructions

Responsible personnel will be trained to identify those instructions to be implemented along with appropriate timetables for implementation.

1.2.2 Organization Level Procedures/Instructions

Organization-level procedures/instructions that further implement the requirements of this instruction are acceptable provided the requirements of this instruction are fully met. The organization procedures/instructions may provide additional detail.

1.2.3 Quality Assurance

The Quality Assurance (QA) requirements are implemented through the use of this overall instruction. This instruction shows how to implement specific QA requirements. Where necessary, direction to other procedures and instructions that must be followed to fully implement the QA requirements is provided.

1.2.3.1 Product and Facility Software Quality Assurance

Product SQA shall be implemented in accordance with the Y60-WP procedures, and facility SQA shall be implemented in accordance with the Y60 procedures. Specific QA requirements are covered by this instruction.

1.2.3.2 Graded Approach

A graded approach to address the SFI Level associated with each software project will be used. This level is based on implications and other considerations including health, safety, and environment. The grading is from SFI Level 3 to Level 0, with Level 3 being the highest level of control and rigor. A graded approach allows resources to be allocated based on the SFI Level, which will provide the greatest economic benefit.

1.2.3.3 Program

Y80-101PD, “Software Management Program Description,” identifies the roles and responsibilities of the personnel responsible for implementing the Software Management Program.

1.2.3.4 Training and Qualification of Personnel

The primary Project Team roles for software with Software Failure Impacts 1, 2, and 3 are to be trained in the application of this instruction. Project Team members may be assigned to multiple Project Team roles.

The primary Project Team roles consist of the Project Manager, System Owner, User Point of Contact, Lead Analyst, Quality Engineer, and the Product Engineer. The Project Manager, Lead Analyst, Quality Engineer, and the Product Engineer are required to have the Software Management Training (Y80-102CL). Y80-102CL is available through Y-12 site training and registration in SAP.

The System Owner and User Point of Contact are required to have training on the Program Description and their roles and responsibilities. Other personnel assigned to a software project due to their project specific expertise may receive this training as determined by the Project Manager.

Training material shall be prepared and presented. Training in the performance of this instruction shall be in accordance with Procedure Y90-027, “Conduct of Training Manual.”

The Project Team develops the required software through implementation of this instruction. The Project Team also generates, as appropriate, training on the software for personnel designated as users.

1.2.3.5 Quality Improvement

Problem detection and prevention shall be accomplished through (1) management assessments as needed, (2) stage exits, and (3) associated problem resolution. Continuous improvement is accomplished through the resolution of deficiencies identified during the problem detection and prevention activities.

Deficiencies identified during the management and independent assessments shall be resolved in accordance with Procedures Y15-312, "Issues Management," and Y15-331, "Lessons Learned Program." For product software, Procedure Y60-WP-160, "Corrective Action System," is used.

1.2.4 Documents and Records

Deliverables are documents that shall be controlled in accordance with Procedure Y15-102, "Document Control." Documentation, whether in the form of paper documents or computer tapes and disks, that become records shall be controlled in accordance with Y15-101, "Records Management," for facility software and Y60-WP-014, "Quality Assurance Records," for product software. Retention for product software is given by DOE-NWS (Nuclear Weapons Schedule 3). Due to the many variations in retention periods for non-product software, Project Teams shall consult with Information Policy, Records & Document Management to establish retention requirements.

1.2.5 Work Processes

1.2.5.1 Manage Software Quality

Software identification is accomplished through the use of a software acronym, a unique identifier composed of up to ten alphanumeric characters.

Software control is accomplished through the implementation of this instruction and is based on the importance of the software to Y-12 operations and the SFI Level.

A graded approach to the software system is taken in accordance with the evaluation results. The software controls include verification, validation, and configuration management.

1.2.5.2 Manage Nonconforming Software

Nonconforming software shall be controlled once it is identified as such. Nonconforming software shall be removed from service if the functionality of the software is nonconforming. Procedure Y60-301, "Control of Nonconforming Items (and Services)," shall be used as appropriate for the nonconforming software. For product software, Procedure Y60-WP-009, "Control of Nonconforming Items," is used to process any product resulting from the use of defective software. The defective software is corrected using the process described in the Maintenance lifecycle stage.

1.2.5.3 Design

The design inputs, process, outputs, verification, and changes of software are built into this instruction. Computer software used to originate or verify design solutions during the design process shall be validated or the status of code validation be identified and documented prior to use in accordance with the implementation of these instructions.

1.2.5.4 Procurement

Software shall be procured and associated procurement documents shall be controlled with the implementation of Procedures Y60-701, "Procurement Quality," Y60-702, "Division Technical Review of Procurement Documents," and Y60-705, "Acquisition, Control, and Traceability of Safety SSCs."

1.2.5.5 Inspection and Acceptance Testing

Inspection and acceptance testing of software is accomplished through the implementation of this instruction, which implements applicable parts of Procedure Y60-801, "Inspection/Test Control."

1.2.5.6 Management Assessment

Management Assessment is accomplished through the stage exits. Additional Management Assessments, performed at the discretion of the Project Manager, shall be performed and documented in accordance with Procedure Y15-902, "Management Assessment."

1.2.5.7 Independent Assessment

Independent assessments, as requested by the Project Manager, shall be performed and documented in accordance with Procedure Y15-903, "Independent Assessment."

1.2.5.8 Security

All identified requirements from Cyber Security relating to software have been incorporated into the life cycle stages. Specific requirements and deliverables are contained in the specific stage section.

1.2.5.9 Software Engineering

All identified Software Engineering requirements have been incorporated into the life cycle stages. Specific requirements and deliverables are contained in the specific stage section.

1.2.5.10 Lifecycle Adaptations

The Project Team is permitted to adapt the project lifecycle as long as fundamental software engineering objectives are retained and security and quality are not compromised. Examples of lifecycle adaptations include repeating, merging, or adding stages, activities, and deliverables. When a lifecycle stage, activity, or deliverable is adapted, the adaptation must be identified, described, and approved by the Project Team in the Software Project Plan. Adaptations should not introduce an unacceptable level of risk into the project or to Y-12 operations. Deliverables that are mandatory, or not subject to exemption under Y80-102, may not be waived via an adaptation of the lifecycle. However, mandatory deliverables may be combined, as specified within the Software Project Plan, with any other deliverables as necessary to meet the project's goals.

Each stage of the software lifecycle normally requires approval of stage completion via a Stage Exit Position Response Form (SEPRF). When stages are combined, the SEPRF deliverable for those stages is also combined, such that signatures on a given set of forms serve as approval for the exit of the combined stages. Likewise, each stage defined in this instruction normally requires revisions to the Software Project Plan, accompanied by approvals of the revisions via the SEPRF. If stages are combined, the Software Project Plan revisions are also combined, such that only one revised Software Project Plan must be produced for the combined stages.

1.2.5.11 Content and Format of Deliverables

This instruction describes various deliverables that must be produced by the Project Team to fulfill the Y-12 SQA Program requirements. For deliverables that are form-based, the standard Y-12 form contained in the Software Application Manager (SAM) should be used when feasible. For deliverables that do not have specific forms or Y-12 implementation guidance, the Project Team shall define the appropriate format, content, and level-of-detail that meets the project's needs and the intent of the deliverable. The DOE SEM contains further suggested guidance on selected deliverables and may be used as a reference by the Project Team. In all cases, the Project Team shall apply appropriate rigor in defining deliverable content, format, and level-of-detail, such that project quality is not compromised and unacceptable risk is not introduced into the project or resulting software.

1.2.6 Special Case/Unique Use Software

1.2.6.1 Electronic Management of Data

Data that software uses to perform functions, analysis or calculation shall be controlled in accordance with this instruction. The data shall be evaluated using the Evaluation Process. The System Owner shall ensure:

- the completeness and accuracy of the data input;
- the completeness and accuracy of subsequent changes to data input;
- that the security of the data is maintained including integrity of the data; and
- that when data is retrieved using a query language, the query shall be checked to ensure it satisfies the Data Owner's requirements for its intended use.

1.2.6.2 Exploratory Projects

This instruction assumes that every project, given enough forethought, can be planned in detail beforehand and can be executed to success. However, some projects involve a large degree of uncertainty and uncontrollable aspects, or involve the need to develop new technology, or simply have never been done before (hence no model exists). In these cases, there is a need to perform preliminary programming of an exploratory nature.

The software must be entered into SAM for inventory purposes. Additionally, it may be in the Project Team's best interest to evaluate the SFI Level of the software assuming it will be implemented.

The Computing and Telecommunications Security Organization (CTSO) must be consulted to determine if the software requires any special security controls.

1.2.6.3 Firmware

Firmware is defined as:

(1) Computer programs and data loaded in a class of memory that cannot be dynamically modified by the computer during processing.

(2) Hardware that contains a computer program and data that cannot be changed in its user environment. The computer programs and data contained in firmware are classified as software; the circuitry containing the computer program and data is classified as hardware. Program instructions are stored in a read-only storage.

(3) An assembly composed of a hardware unit and a computer program integrated to form a functional entity whose configuration cannot be altered during normal operation. The computer program is stored in the hardware unit as an integrated circuit with a fixed logic configuration that will satisfy a specific application or operational requirement.

Firmware that is developed or modified for use at Y-12 shall be entered into SAM, categorized as to the software failure impact level and comply with the requirements for the approved SFI level. This software may be handled as a part of an overall system.

1.2.6.4 Measuring and Test Equipment (M&TE) Calibration

M&TE that contains software, firmware or programmable hardware shall be calibrated, adjusted, and maintained as a unit at prescribed intervals, or prior to use, against reference calibration standards having traceability to nationally recognized standards. If no nationally recognized standards or physical constants exist, the basis for calibration shall be documented. The calibration documentation shall show both the M&TE identification and the software identification (version, date, etc.). Reference Y60-802, "Calibration and Control of Measuring and Test Equipment (M&TE)," for further information.

M&TE software that is developed or modified for use at Y-12 shall be entered into SAM, categorized as to the software failure impact level and must comply with the requirements for the approved SFI level.

1.2.6.5 Numerical Control (NC) Programs

An NC part program is a set of instructions whose primary function is to control the motion and ancillary functions of a fabrication or inspection tool for a specific application. NC part programs are exempt from the purview of this instruction. Software that generates, translates, or executes numerical control part programs is subject to the requirements of this instruction. Numerical control part programs and operator instructions used to certify weapons product in the Dimensional Metrology/Inspection Department of the Product Certification Organization are subject to the requirements of Weapon Administrative Product Specification, OO-A-108.

1.2.6.6 Programmable Logic Controller (PLC) Ladder Diagrams

Ladder diagrams tell a programmable controller how to sequence through a series of steps such as starting or stopping a process, turning voltages on and off, reading data, downloading data, detecting alarm conditions, and transmitting alarms. PLCs that are used directly in production may have considerable product and safety consequences. Ladder diagrams are "programs" in the same sense that an ordinary FORTRAN program can be represented as a logic flow diagram. PLCs shall meet the requirements of this instruction.

1.2.6.7 Software Routines and Macros

Many COTS packages include the ability to write custom macro routines for local use. For example, Microsoft Excel permits the user to develop custom macros to augment its basic capabilities. The use of standard features in a COTS package is not considered programming. However, the development of custom macros to produce custom-programmed applications is considered software under Y80-102 and the following requirements shall apply.

For software routines and macros written for use with COTS software, whose results can be verified by visual inspection or hand calculations, the user shall be responsible for ensuring the software results are correct for the specified range of input parameters. Software routines and macros written for use with COTS software that may not be verified by visual inspection and/or hand calculations shall be categorized as to the software failure impact level and must comply with the requirements for the approved SFI level.

1.2.6.8 Freeware/Open Source

Freeware or Open Source software is not permitted for use in a classified environment without prior approval from the Cyber Security Program Manager. In the unclassified environment, freeware is permitted after configuration for local use. For unclassified use of freeware, CTSO should be contacted for further guidance. In all cases, the use of freeware is governed by this instruction and shall be handled in a manner consistent with that defined for developed externally software.

1.2.6.9 Operating Systems and Related Products

Operating systems and related products that are controlled under Procedures Y15-405PD, “Information Technology Configuration Management Program,” Y15-406INS, “Information Technology Configuration Management Instruction,” and Y19-401INS, “Automated Information System (AIS) Security Handbook,” may have requirements of this instruction waived as deemed appropriate by the SQA Manager, Cyber Security Program Manager, and Weapons Product Quality Manager. To request a waiver, the Project Team submits a waiver proposal to the SQA Manager. The SQA Manager, the Cyber Security Program Manager, and the Weapons Product Quality Manager must all concur and approve the request. The approved list will be maintained by the SQA Manager.

1.2.6.10 Common Desktop Productivity Tools

Common desktop applications used within Y-12 may be exempted from the requirements of this instruction. Internally developed custom macros must adhere to the requirements of Section 1.2.6.7. Approval by the SQA Manager and the Cyber Security Program Manager is required for software to be placed into this category. The list is available from the Desktop Computing Home Page.

1.2.6.11 Limited Use of Software Prior to Acceptance

Situations may arise that require limited, controlled use of software prior to completion of all SQA tasks in this instruction. The SQA Manager will consider the need on a case by case basis. The SQA Manager will consult with the Weapons Product Quality Manager before authorizing use of product software prior to acceptance. At a minimum, the software must have an approved SFI, some test documentation for the parts of the software that will be used, assessment of the risk associated with the use, a plan that details how use of the software will be controlled, and the quality checks that will be used to verify results before approval can be granted. Depending upon the use of the software, other requirements may apply.

1.2.7 Roles and Responsibilities

Roles, responsibilities, and tasks pertinent to each life cycle stage are identified in the lifecycle stage in Section 2.

2 Life Cycle Stages

The Life Cycle Model used for the Y-12 Software Management Program consists of nine major stages, with each stage consisting of one or more activities that may be further divided into tasks. Life Cycle stages, activities, and tasks have deliverable work products and measurable endpoints. The nine life cycle stages, activities, tasks, and work products are discussed in the following sections. Multiple stages between the Planning Stage and the Installation and Acceptance Stage may be worked by the Project Team concurrently or combined as documented in the Software Project Plan.

It is recommended that the Project Team complete the Planning Stage prior to moving into other stages. During the Planning Stage, the Software Failure Impact (SFI) is determined and approved by the SRB. Additionally, the Project Team determines any Lifecycle Adaptations (Section 1.2.5.10) and which Project Team exemptions will be approved.

Several activities are common to each life cycle stage. The input needed to perform the activities and tasks should be available before initiating a stage. At the discretion of the Project Manager, assessments may be conducted during each stage to assess the quality of the deliverables, security, state of development, and project status. The final activity in each life cycle stage except for the Retirement Stage is the Stage Exit, where the System Owner, Project Manager, and other project stakeholders assess the quality of the deliverables, security, state of development, and project status. The result of this assessment is the determination and documentation of whether to close the stage, continue with the current stage(s), return to a previous stage, or terminate the project. If the decision is to close the stage, then the Stage Exit Form is approved indicating completion of the deliverables for the stage. When multiple stages are worked concurrently, one Stage Exit will cover all deliverables for the combined stages but this must be documented in the Software Project Plan.

Careful consideration should be given to purchasing COTS software prior to expending time, resources, and costs associated with in-house custom-built systems. Initiation for the procurement of any hardware or software is encouraged to occur well in advance of the planned need for these products. The Project Team is responsible for all of the procurement, installation, and testing responsibilities.

2.1 Planning Stage

2.1.1 Purpose

The purpose of the Planning Stage is to initiate and plan software projects for Y-12. Requirements and guidelines for documenting the activities of the Planning Stage are explained in this instruction. Planning includes collecting information for software application identification, screening, and software failure impact (SFI) level determination in preparation for writing the Software Project Plan.

2.1.2 Strategy

The Planning Stage is usually the first stage in the life cycle of a software project. In this stage, the general system requirements are identified, the users' environment is analyzed, the project objectives and scope are defined, the feasibility of the project is determined, and the Software Project Plan is developed.

Project planning applies to all projects regardless of size or SFI. Planning involves selecting the strategies, policies, programs, and procedures for achieving the objectives and goals of the project. Planning is deciding what to do, how to do it, when to do it, where to do it, and who will do it.

The requirements identified in project-related materials (e.g., a business case) are the primary input to the Software Project Plan. The level of detail will vary with project size. The preparation of the Software Project Plan and related materials involves critical planning issues: the identification of preliminary requirements; staff, schedule, and cost estimates; the technical and managerial approaches that will be used; and the assessment of potential risks associated with the project.

During this stage, the System Owner and users are interviewed to identify their business needs and expectations for the product, to gain a common understanding of the task assignment, and to determine how the project supports the DOE and organizational missions and long-range information resource management plans. The System Owner represents the organizational unit that is funding the project, and users are the DOE employees and contractors who will use the product. In this stage, the Project Team should be focused on identifying what the project will automate, and whether developing an automated solution makes sense from business, cost, and technical perspectives. If the project is feasible, then estimates of time, cost, and resources must be formulated for the project, and risk factors must be assessed. It is important for the Project Team to work closely with representatives from all functional areas that will be involved in providing resources, information, or support services for the project. The information that is gathered in this stage is used to plan and manage the project throughout its lifecycle.

Evaluation is the other core activity in the Planning Stage. Evaluation of software is based on the use of the software, the source of the software, and the consequence of failure. More rigorous control is required for projects that involve greater risk.

Information necessary to identify and manage a software project will be collected at the initiation of software acquisition or development and input into the software inventory system using the Software Application Manager (SAM). The information will be added to or revised as the application definition is established and requirements are implemented. SAM provides dynamic links to associated reference materials, instructions, documentation, and templates. SAM is accessible from the Y-12 Home Page - YSOURCE.

2.1.3 Stage Deliverables, Exemptions and Approvals

This section describes various deliverables that must be produced by the Project Team to fulfill the Y-12 SQA Program requirements. For deliverables that are form-based, the standard Y-12 form contained in SAM should be used when feasible. Templates for each deliverable are available from SAM and are used at the discretion of the Project Team. If the standard Y-12 form or template is not used by the Project Team, the team shall define the appropriate format, content, and level-of-detail that meet the project's needs and the intent of the deliverable. The DOE SEM contains additional guidance on selected deliverables and may be used as a reference by the Project Team. In all cases, the Project Team shall apply appropriate rigor in defining deliverable content, format, and level-of-detail, such that project quality is not compromised and unacceptable risk is not introduced into the project.

The Deliverables Matrix (Table 1) defines the graded approach based on the software source and software failure impact for implementing the Planning Stage at Y-12. Some deliverables are not exemptible and are marked "NE." Other deliverables are not exemptible under certain conditions and are marked "NEx" where "x" denotes the condition. When the software does not meet the specified condition, the deliverable is optional. Deliverables that may be exempted by the SRB provided adequate justification is provided are marked "SB." Deliverables that may be exempted by the Project Team provided adequate justification is documented are marked "PT." Optional deliverables are shaded in the table.

Responsibility for the approval of the specific deliverable is defined in Section 2.1.4.

Table 1 - Deliverables Matrix for the Planning Stage

Software Source	COTS				DEV INT				DA/GA				DEV EXT			
SFI	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**
Software Inventory	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE
Software Project Plan		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE
Feasibility Study Document						PT	SB	SB						PT	SB	SB
Protected Software Identification Form		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE
Mission-Essential Evaluation - Continuity of Operations Statement		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1
Risk Assessment		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1
Stage Exit Position Response Form		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE

NE – No exemption.
NE1 - Required for Protected Software.
NE2 – Required for Mission-essential Software.
NE3 – Required when software is developed for the classified environment by persons without a Q-clearance.
NE4 – Required when software will be installed in multiple environments.
SB – Required unless the SRB approves the exemption request.
PT – Required unless the Project Team approves the exemption request.

2.1.3.1 Software Inventory

The software inventory entry reflects information regarding a specific software project. The information includes system requirements, staff roles (e.g., system owner, lead analyst), security certification, SFI evaluation, and any other items currently defined within the software inventory repository. SAM is used at Y-12 for inventorying all software covered by this instruction. The software inventory will be entered directly into SAM accessible from YSOURCE.

2.1.3.2 SFI Evaluation

A software failure impact is a categorization of software, which determines the applicable requirements, configuration management, and approvals required within each of the life cycle stages. The SFI evaluation is documented in the software inventory deliverable. The categorization of software by its failure impact level requires the Project Team to evaluate the consequences of failure of a given system in a variety of areas, including health and safety, the environment, Y-12 operations, and Y-12 financial and programmatic goals. Table 2 summarizes the approach to failure impact categorization of software at Y-12. There are four failure impact levels, numbered 0 through 3 in increasing order of failure impact. Software at Y-12 shall be evaluated for failure impact as part of project planning. Software must be assigned to the highest failure impact for the threshold met.

Software shall be evaluated and graded by its source and consequence of failure. The SFI for Protected software shall be a minimum of 1. Mission-essential software shall have an SFI of 3. Health, safety, and environmental software shall default to SFI Level 3 unless written rationale and justification is provided for a lower SFI Level. Product software shall default to SFI Level 2 unless written rationale and justification is provided for assigning a SFI Level of 1. Product software assignment to SFI 1 must be approved by the SRB.

The failure consequences listed in Table 2 are intended to be a summary of items to be considered when evaluating the possibility of a failure of the system in question. The table is intended to provide guidance to a given Project Team when determining the failure impact. It is the responsibility of the Project Team under this instruction to evaluate a particular project and make an appropriate determination of the failure impact, which may include consideration of other issues beyond those listed.

For all assessments of failure impact, the Project Team may wish to document assumptions, evaluations of risk or other parameters, and related data as part of the project record.

When assessing failure impact, the Project Team must consider the likelihood of a given set of failures, effects on other systems or projects that interface from or to the application in question, and possible time to recover and/or repair a given failure. If the direct impact on other interfaces is minimal or far removed, those factors shall be given appropriate weight in making a determination. Likewise, if a given amount of downtime is acceptable for repairs, or if alternate manual or other processes may temporarily substitute for the system in question, that should also be considered. These are only a limited set of the "external" factors that must be considered when assessing failure impact. The Project Team must take a holistic view of the system and its ability to impact Y-12 via a failure, since it is not possible to prepare a cookbook approach to this assessment.

During a project life cycle, the failure impact assessment may be revised upon the Project Team's decision, as more information regarding the project is determined or as scope changes. The Software Review Board must approve all SFI Level Changes. If at any time during the project the failure impact level changes, all requirements for the new software failure impact level must be met (including those from prior stages) unless an exemption is granted by the SRB or the Project Team, as authorized.

Written rationale and justification is not required to increase the SFI Level to a higher SFI Level than the default. Additional responses to questions presented to the requester will provide a determination of the SFI of the software application. Misuse and abuse of the "grading criteria" will be controlled through the SRB review and approval process, employee training, and independent assessments.

Table 2 - Software Failure Impact Threshold Criteria

Failure Impact	Description	Consequences
3	Severe impact	Immediate or eventual loss of life or disability for employees or the public or an immediate failure of safety or other required operating controls Catastrophic or severe damage to the ecological system, including release of hazardous materials or contamination Excessive exposure to hazardous materials for employees or the public Failure to meet facility-wide mission or mandates, including physical/information/cyber security, financial obligations, critical manufacturing goals, or breakdown of key operations Loss or disabling of Y-12 production capability Non-compliance with federal, state, or local laws, including loss or corruption of information or data Severe failure of analytical support of environmental, safety, nuclear criticality, or other required operating controls
2	Significant impact	Failure to meet award fee milestones Significant damage to the ecological system, excluding release of hazardous materials or contamination Personnel exposure to hazardous materials below legal or recommended maximum limits Significant increased cost to business unit Significant negative impact on ability to accomplish mission by impairment of key Y-12 operations Lost time injury requiring medical treatment or significant violations of required operating controls, including safety Significant impact spanning multiple organizations or projects, including an impact on other systems via interfaces Significant failure of analytical support of environmental, safety, nuclear criticality, or other required operating controls
1	Moderate impact	Moderate increased cost to business unit – no violation of authorized budgetary constraints Moderate disruption to other systems via interfaces Moderate impact on ability to accomplish organizational or project goals Moderate impact on work processes – alternative systems/methods exist to perform work or moderate system downtime is acceptable Moderate failure of analytical support of environmental, safety, nuclear criticality, or other required operating controls
0	Minor impact	Personal inconvenience Minor financial or business impact Minor disruption to work processes Minor failure of analytical support of environmental, safety, nuclear criticality, or other required operating controls

2.1.3.3 Software Project Plan

The Software Project Plan (SPP) documents the project scope, cost, schedule and other project related information. It is the responsibility of the Project Manager to ensure that the SPP accurately reflects the current status of the overall project.

The SPP must contain the following information:

- Project tasks to be performed, required resources, estimated/actual task start and completion dates, constraints/parameters affecting the schedule
- Any life cycle adaptations
- Elements and content of a software quality plan including change management through the Installation and Acceptance Stage and software error reporting and correction during integration testing and acceptance testing
- Process for software error reporting and correction through the Installation and Acceptance Stage
- Project risk management
- Method for Requirements Traceability including validation and verification through the Installation and Acceptance Stage
- Other project-related information such as:
    - Statement of Project Scope
    - Statement of Project Objectives
    - Description of User Environment
    - Statement of High Level Requirements
    - Functional Area Contact List & Project Profile

2.1.3.4 Feasibility Study

The Feasibility Study Document documents the feasibility and cost benefit of successfully developing and implementing the project. Software and hardware alternatives are reviewed and used to formulate preliminary platform options. Project feasibility leads to a "go" or "no go" decision about the project. Determining project feasibility is an interactive process of collecting and analyzing data and searching for cost-effective, viable technical solutions. In developing the Feasibility Study Document, the Project Team uses the project objectives, scope, and high-level requirements as the basis for determining project feasibility.

2.1.3.5 Protected Software Identification (PSI) Form

The PSI form documents the determinations of classification, sensitivity, essentiality, and protected software. Classification and UCNI determinations must be made by an Authorized Derivative Classifier and an UCNI Reviewing Official respectively. The PSI form is available through the CTSO Home Page and SAM.

2.1.3.6 Mission-Essential Evaluation/Continuity of Operations Statement

The Mission-Essential Evaluation allows the determination of the impact on Y-12's mission in the event of software failure or a hardware environment failure. The evaluation form corresponds to information outlined in the Automated Information System (AIS) Security Handbook. The Mission-Essential Evaluation form contains the criteria for determining if the software is mission-essential. This form is available via SAM. Based on the result of the local mission-essential evaluation, a Continuity of Operations Statement may be required.

For non-mission-essential software products, the Continuity of Operations Statement documents the decision that the Continuity of Operations Test Plan is not required. Alternate measures to meet the business requirements are defined in the Application Security Specification and Controls. The Continuity of Operations Statement is available from SAM.

2.1.3.7 Risk Assessment

The Risk Assessment is performed as part of establishing the overall security profile of a protected application. The purpose is to assess and quantify risks associated with the application. Based on this assessment, certain protection mechanisms may be implemented as part of the application to minimize the associated risks. The initial assessment performed during the Planning Stage is a high level effort to determine what future specific security actions will be required for the application. (Reference Y19-401INS, Chapter 3.)

2.1.3.8 Stage Exit Position Response Form (SEPRF)

The SEPRF is the tool for obtaining approval of the stage deliverables and to move forward into the next stage of development or implementation. Preliminary work may be performed on the next scheduled stage(s) if documented in the SPP while reviews and approvals are being obtained to close the current stage officially. The roles and responsibilities for each stage identify the deliverables that each reviewer is approving or disapproving. An electronic form and a form template are available from SAM.

When an approved deliverable from a previous stage is modified in the current stage, such as updating the SPP, the SEPRF should note the approval of the updated deliverable. The designated approvers from the previous stage will approve the updates in the current stage.

2.1.3.9 Deliverable Approvals

Approvals of deliverables are based on the documents being reviewed and comments being resolved with responsible personnel prior to approval. Approval responsibility is assigned to the roles identified Section 2.1.4. Approval of the stage deliverables is done via approval of the SEPRF except as noted.

2.1.3.10 Project Team Exemption Requests

For exemption requests that require disposition from the Project Team, the exemption rationale shall be documented. The documentation may be in the SPP, by an Exemption Request Form, or in the comments field for the deliverable in SAM.

2.1.3.11 SRB Exemption Requests

For exemption requests that require disposition from the SRB, an Exemption Request Form or SAM Exemption Request must be submitted to the SRB. Exemption requests submitted to the SRB shall contain a statement of the deliverable that is being exempted and the rationale for the exemption. SRB exemption approvals/rejections shall be returned to the requester. The SRB minutes shall contain documentation of the exemption action.

2.1.4 Roles and Responsibilities in the Planning Stage

The Project Manager has the overall responsibility for the successful execution of the Planning Stage. This instruction does not assign the completion of deliverables to specific Project Team members. The Project Manager assigns the completion of deliverables to the appropriate Project Team members. The roles and responsibilities listed below identify the role normally responsible for the action. Project Team members can be assigned multiple project roles by the Project Manager. Where multiple roles are listed as creating a deliverable, it is expected that these roles will work together to create the deliverable. Responsibilities other than approval responsibilities may be re-assigned to meet project needs.

2.1.4.1 System Owner

The System Owner (SO) must be a BWXT Y-12 employee and identifies the functional characteristics of the software. Responsibilities in the Planning Stage include:

- Assures budgeting that is adequate for development or acquisition and testing to ensure that the requirements of this instruction are met and that the software meets its established requirements.
- Accountable for the software accuracy and the necessary documentation that is needed to support the software.
- Ensuring that applicable Y-12 policies and procedures not specifically addressed in this instruction shall be implemented (e.g., maintenance of licensing agreements).
- Reviews and concurs with the Software Evaluation and the failure impact assigned.
- Establishes a funding source for costs associated with the project.
- Oversees the identification and selection of a Project Manager.
- Generates Protected Software Identification (PSI) form and approves via a signature on the document and submits it to CTSO.
- IF protected software, THEN prepares the Mission-Essential Evaluation and Continuity of Operations Statement to document whether the application is mission-essential and approves it via a signature on the document.
- Approves the following deliverables via the Stage Exit Position Response Form:
    - Feasibility Study Document
    - Software Project Plan
    - Risk Assessment

2.1.4.2 Project Manager

The Project Manager provides overall leadership of the project and assures successful execution of this instruction. The Project Manager may be selected from any participating organization on the basis of his or her knowledge of the subject area, leadership skills, etc. Responsibilities in the Planning Stage include:

- Responsible for the selection and approved participation of appropriate Project Team members.
- Assigns development of various stage deliverables and project activities to the appropriate Project Team member depending upon the team member's knowledge of the subject, writing skills, available time, etc.
- Ensures that roles and responsibilities of Project Team members are identified and understood.
- Establishes and maintains the Project Files for required project information through the Installation and Acceptance Stage.
- Selects the software system acronym and name and enters software project information into SAM.
- Determines and documents software use, software type, and software source.
- Identifies the initial software failure impact based on the definition (see Table 2, Software Failure Impact Threshold Criteria) based upon the consequences of failure and documents the SFI by answering the appropriate questions in SAM.
- Submits the SFI to the SRB for approval.
- Assists the System Owner with generating the Protected Software Identification (PSI) form.
- IF protected software, THEN works with the CTSO to determine whether the Lead Analyst will serve as the CTSO Representative.
- Assists the Product Engineer (PE) and Quality Engineer (QE) in the product software determination.
- Drafts the Software Project Plan.
- Documents Project Team decisions regarding PT exemptions of deliverables.
- Submits request for SB exemption via SAM, when feasible, or the Exemption Request Form.
- IF an Exemption Request Form is used for an exemption, THEN retains the approved or rejected exemption request in the project file.
- Approves the following deliverables via the Stage Exit Position Response Form:
    - Feasibility Study Document
    - Software Project Plan
    - Risk Assessment

2.1.4.3 Project Team

The Project Team provides support to project activities under the direction of the Project Manager. Responsibilities in the Planning Stage include:

- Identifies exemptions of deliverables using criteria in these instructions.
- Determines project feasibility and documents in the Feasibility Study deliverable.
- Identifies, documents, and maintains communications with appropriate functional areas.
- Ensures that risk is properly documented in the Risk Assessment.
- Develops the Software Project Plan.
- Incorporates the appropriate QA activities for the project into the Software Project Plan.
- Distributes the Software Project Plan to appropriate personnel for review and comment.
- Concurs with PT deliverable exemptions.
- Concurs with SB deliverable exemption requests prior to the Project Manager submitting the request to the SRB.
- Coordinates with the Information Management Systems (IMS) Organization on maintaining the required records and the associated record retention requirements.
- Ensures that the classification determinations for the information and application are made by an Authorized Derivative Classifier (ADC) and documented on the PSI form.
- Reviews and adjusts, if necessary, the software failure impact at the end of each stage up to the Installation and Acceptance Stage. Inputs updated information to the software inventory system via SAM.
- Prepares the Stage Exit Position Response Form.

2.1.4.4 User Point of Contact (POC)

The User POC represents the needs and interests of the users of the system and ensures that the system requirements are defined and met. Responsibilities in the Planning Stage include:

- Ensures that the sensitivity of the data in the system is determined by the owner(s) of the data.
- Ensures that the needs of users are understood and met through the system as delivered.
- Approves the Software Project Plan via the Stage Exit Position Response Form.

2.1.4.5 SQA Manager

The SQA Manager provides oversight of the Software Management Program and guidance to Project Teams on the implementation of requirements in this instruction. Responsibilities in the Planning Stage include:

- Identifies Software Project Plans to be reviewed by the SRB.
- Assigns a QE to the Project Team to ensure that appropriate QA activities are addressed in the Software Project Plan, and to perform other QA related activities throughout the software stages.

2.1.4.6 Computing and Telecommunications Security Organization (CTSO)

CTSO provides leadership in the protection of information technology. Responsibilities in the Planning Stage include:

- Provides guidance and assists in determining if software is protected software.
- Reviews the (PSI) form and notifies the System Owner if there is a disagreement with the protected software determination.
- Reviews the mission essentiality determination.
- Reviews other project deliverables for security implications at the discretion of the Cyber Security Program Manager.
- IF the Lead Analyst is not approved as the CTSO Representative, THEN notifies the Project Manager who will be the CTSO Representative.
- Reviews project information in the software inventory system.
- Concurs with the protected software determination via the Stage Exit Position Response Form.
- Approves the Risk Assessment via the Stage Exit Position Response Form.

2.1.4.7 CTSO Representative

The CTSO Representative assures the appropriate cyber security requirements are incorporated into the software if protected. The CTSO Representative defaults to the Lead Analyst. Responsibilities in the Planning Stage include:

- Interfaces with CTSO on cyber security matters associated with the project; and
- Approves the Software Project Plan via the Stage Exit Position Response Form.

2.1.4.8 Software Review Board (SRB)

The SRB assures proper characterization of the software by reviewing and approving/rejecting the SFI and product determination. Responsibilities in the Planning Stage include:

- Reviews selected documentation as requested by the SQA Manager for compliance with the Y80-102 instruction and provides comments to the QE assigned to the software project.
- Reviews the Software Inventory information in SAM.
- Approves/rejects the determination of the Software Failure Impact that results from the evaluation by the Project Team.
- IF the Project Team and the SRB do not agree on the SFI, THEN the SRB determines the software failure impact.
- Approves/rejects SB exemption requests for identified required deliverables.
- IF an Exemption Request form is used to request an exemption, THEN assigns an identification number to the exemption request.
- Reviews and approves or rejects requested exemptions, with rejection rationale.
- IF the Exemption Request form is used, THEN the exemption approval/rejection is marked, a copy is placed in the SRB files and the exemption request is returned to the Software Owner and the Project Manager.

2.1.4.9 Lead Analyst

The Lead Analyst provides technical direction for the software design and implementation activities. Responsibilities in the Planning Stage include:

- Provides input on the Mission-Essential Evaluation and Continuity of Operations Statement.
- Approves the following deliverables via the Stage Exit Position Response Form:
  - Feasibility Study Document
  - Software Project Plan

2.1.4.10 Quality Engineer (QE)

The QE provides independent oversight and review of the project deliverables. Responsibilities in the Planning Stage include:

- Ensures the Software Project Plan incorporates QA elements.
- Assists in determining if the application is product software.
- Reviews the software failure impact evaluation.
- Approves Software Project Plan via the Stage Exit Position Response Form.

2.1.4.11 Product Engineer (PE)

The PE assures appropriate SQA activities are performed and product related software is managed consistent with QC-1. Responsibilities in the Planning Stage include:

- Assists in determining if the application is product software.
- Reviews the software failure impact evaluation.
- IF Product Software, THEN approves Software Project Plan via the Stage Exit Position Response Form.

2.2 Requirements Definition Stage

2.2.1 Purpose

To establish a process for identifying and documenting the requirements for application software projects for Y-12. The primary purpose of the Requirements Definition Stage is to develop a mutual understanding of system requirements between the Project Team, the System Owner, and the user community.

2.2.2 Strategy

This section presents the deliverables and responsibilities for the Requirements Definition Stage.

2.2.3 Stage Deliverables, Exemptions and Approvals

This section describes various deliverables that must be produced by the Project Team to fulfill the Y-12 SQA Program requirements. For deliverables that are form-based, the standard Y-12 form contained in SAM should be used when feasible. Templates for each deliverable are available from SAM and are used at the discretion of the Project Team. If the standard Y-12 form or template is not used by the Project Team, the team shall define the appropriate format, content, and level-of-detail that meet the project's needs and the intent of the deliverable. The DOE SEM contains additional guidance on selected deliverables and may be used as a reference by the Project Team. In all cases, the Project Team shall apply appropriate rigor in defining deliverable content, format, and level-of-detail, such that project quality is not compromised and unacceptable risk is not introduced into the project.

The Deliverables Matrix (Table 3) defines the graded approach based on the software source and software failure impact for implementing the Requirements Definition Stage at Y-12. Some deliverables are not exemptible and are marked "NE." Other deliverables are not exemptible under certain conditions and are marked "NEx" where "x" denotes the condition. When the software does not meet the specified condition, the deliverable is optional. Deliverables that may be exempted by the SRB provided adequate justification is provided are marked "SB." Deliverables that may be exempted by the Project Team provided adequate justification is documented are marked "PT." Optional deliverables are shaded in the table.

Responsibility for the approval of the specific deliverable is defined in Section 2.2.4.

Table 3 - Deliverables Matrix for the Requirements Definition Stage

Software Source	COTS				DEV INT				DA/GA				DEV EXT			
SFI	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
Software Requirements Specification		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE
Stage Exit Position Response Form		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE

NE – No exemption.
NE1 - Required for Protected Software.
NE2 – Required for Mission-essential Software.
NE3 – Required when software is developed for the classified environment by persons without a Q-clearance.
NE4 – Required when software will be installed in multiple environments.
SB – Required unless the SRB approves the exemption request.
PT – Required unless the Project Team approves the exemption request.

2.2.3.1 Software Requirements Specification

The Software Requirements Specification describes the manual and automated requirements for a software product in the user's language. It includes the system's functional requirements, performance requirements, input and output requirements, communication requirements, security and access requirements, identification and grouping of data elements, data backup and recovery requirements, implementation requirements, and interface requirements. It identifies what the software product must do to support the system owner and/or users' business functions and objectives. Processing requirements for the software product are documented without implying how the system will provide those functions. All design constraints are documented and data constraints identified and defined.

2.2.3.2 Stage Exit Position Response Form (SEPRF)

The SEPRF is the tool for obtaining approval of the deliverables to move forward into the next stage of development or implementation. Preliminary work may be performed on the next scheduled stage(s) if documented in the SPP while reviews and approvals are being obtained to close the current stage officially. The roles and responsibilities for each stage identify the deliverables that each reviewer is approving or disapproving. An electronic form and a form template are available from SAM.

When an approved deliverable from a previous stage is modified in the current stage, such as updating the Software Project Plan, the SEPRF should note the approval of the updated deliverable. The designated approvers from the previous stage will approve the updates in the current stage.

2.2.3.3 Deliverable Approvals

Approvals of deliverables are based on the documents being reviewed and comments being resolved with responsible personnel prior to approval. Approval responsibility is assigned to the roles identified in Section 2.2.4. Approval of the stage deliverables is done via approval of the SEPRF.

2.2.3.4 Project Team Exemption Requests

For exemption requests that require disposition from the Project Team, the exemption rationale shall be documented. The documentation may be in the SPP, by an Exemption Request Form or in the comments field for the deliverable in SAM.

2.2.3.5 SRB Exemption Requests

For exemption requests that require disposition from the SRB, an Exemption Request Form or SAM Exemption Request must be submitted to the SRB. Exemption requests submitted to the SRB shall contain a statement of the deliverable that is being exempted and the rationale for the exemption. SRB exemption approvals/rejections shall be returned to the requester. The SRB minutes shall contain documentation of the exemption action.

2.2.4 Roles and Responsibilities in the Requirements Definition Stage

The Project Manager has the overall responsibility for the successful execution of the Requirements Stage. The Y80 Procedure does not assign the completion of deliverables to specific Project Team members. The Project Manager assigns the completion of deliverables to the appropriate Project Team members. The roles and responsibilities listed below identify the role normally responsible for the action. Project Team members can be assigned multiple project roles by the Project Manager. Where multiple roles are listed as creating a deliverable, it is expected that these roles will work together to create the deliverable. Responsibilities other than approval responsibilities may be re-assigned to meet project needs.

2.2.4.1 System Owner

Responsibilities in the Requirements Definition Stage include:

- Provides guidance to the Project Manager as to the expected direction and duration of this stage.
- Establishes or adjusts the project's ranking and priority in relation to other projects competing for resources.
- Resolves conflicts over scope and requirements statements.
- Provides input to the Software Requirements Specification.
- Approves the Software Requirements Specification via the Stage Exit Position Response Form.

2.2.4.2 Project Manager

Responsibilities in the Requirements Definition Stage include:

- Initiates any required hardware and software procurement activities and tracks progress.
- Provides input to the Software Requirements Specification, including ensuring that all requirements-related issues are resolved either by incorporation into the specification or an explicit determination of non-applicability, non-cost-effectiveness, or other reasons for non-inclusion in the specification.
- Tracks and updates project progress against the Software Project Plan.
- Documents Project Team decisions regarding exemptions of Y80-102 deliverables.
- Submits a request for exemption via SAM, when feasible, or the Exemption Request Form.
- IF an Exemption Request Form is used for an exemption, THEN retains the approved or rejected exemption request in the project file.
- Approves the Software Requirements Specification via the Stage Exit Position Response Form.

2.2.4.3 Project Team

Responsibilities in the Requirements Definition Stage include:

- Initiates any required hardware and software procurement activities and tracks progress.
- Provides input to the Software Requirements Specification, including ensuring that all requirements-related issues are resolved either by incorporation into the specification or an explicit determination of non-applicability, non-cost-effectiveness, or other reasons for non-inclusion in the specification.
- Provides project status to the Project Manager.
- Concurs with PT deliverable exemptions.
- Concurs with SB deliverable exemption requests prior to the Project Manager submitting the request to the SRB.

2.2.4.4 User Point of Contact

Responsibilities in the Requirements Definition Stage include:

- Provides input to the Software Requirements Specification regarding functionality and related requirements.
- Approves the Software Requirements Specification via the Stage Exit Position Response Form.

2.2.4.5 Computing and Telecommunications Security Organization (CTSO)

In the Requirements Definition Stage, CTSO reviews project deliverables for security implications at the discretion of the Cyber Security Program Manager.

2.2.4.6 CTSO Representative

Responsibilities in the Requirements Definition Stage include:

- Provides input to the Software Requirements Specification regarding security requirements.
- For protected software, approves the Software Requirements Specification via the Stage Exit Position Response Form.

2.2.4.7 Software Review Board (SRB)

Responsibilities in the Requirements Definition Stage include:

- Approves/rejects exemption requests for identified required deliverables.
- IF an Exemption Request form is used to request an exemption, THEN assigns an identification number to the exemption request.
- Reviews and approves or rejects requested exemptions, with rejection rationale.
- IF the Exemption Request form is used, THEN the exemption approval/rejection is marked, a copy is placed in the SRB files and the exemption request is returned to the Project Manager.

2.2.4.8 Lead Analyst

Responsibilities in the Requirements Definition Stage include:

- Provides input to the Software Requirements Specification, including hardware, software, performance, and related issues.
- Prepares the Software Requirements Specification.
- Approves the Software Requirements Specification via the Stage Exit Position Response Form.

2.2.4.9 Records Analyst

The Records Analyst assures the appropriate Records Management and Document Control requirements are met for the project. Responsibilities in the Requirements Definition Stage include:

- Provides requirements for inclusion in the Software Requirements Specification regarding Record Retention and Document Control.
- Ensure DOE-Y-12 Contract Clauses, DOE-STD-4001-2000, “Design Criteria Standard for Electronic Records Management Software Applications,” and DOE O 200.1-1, “Information Management Program (36CFR, 1234),” requirements are incorporated in the application.
- Approves the Software Requirements Specification via the Stage Exit Position Response Form.

2.3 Functional Design Stage

2.3.1 Purpose

To construct functions from the requirements identified in the Requirements Definition Stage that define the software work product to the level of detail necessary to build the system design. Requirements and guidelines for documenting the activities of the Functional Design Stage are explained in this section.

2.3.2 Strategy

This section presents the Deliverables and Roles and Responsibilities for the Functional Design. A graded approach based upon software source and software failure impact is used.

2.3.3 Stage Deliverables, Exemptions and Approvals

This section describes various deliverables that must be produced by the Project Team to fulfill the Y-12 SQA Program requirements. For deliverables that are form-based, the standard Y-12 form contained in SAM should be used when feasible. Templates for each deliverable are available from SAM and are used at the discretion of the Project Team. If the standard Y-12 form or template is not used by the Project Team, the team shall define the appropriate format, content, and level-of-detail that meet the project's needs and the intent of the deliverable. The DOE SEM contains additional guidance on selected deliverables and may be used as a reference by the Project Team. In all cases, the Project Team shall apply appropriate rigor in defining deliverable content, format, and level-of-detail, such that project quality is not compromised and unacceptable risk is not introduced into the project.

The Deliverables Matrix (Table 4) defines the graded approach based on the software source and software failure impact for implementing the Functional Design Stage at Y-12. Some deliverables are not exemptible and are marked "NE." Other deliverables are not exemptible under certain conditions and are marked "NEx" where "x" denotes the condition. When the software does not meet the specified condition, the deliverable is optional. Deliverables that may be exempted by the SRB provided adequate justification is provided are marked "SB." Deliverables that may be exempted by the Project Team provided adequate justification is documented are marked "PT." Optional deliverables are shaded in the table.

Responsibility for the approval of the specific deliverable is defined in Section 2.3.4.

Table 4 - Deliverables Matrix for the Functional Design Stage

Software Source	COTS				DEV INT				DA/GA				DEV EXT			
SFI	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**
Functional Design Document						PT	SB	SB						PT	SB	SB
Stage Exit Position Response Form						NE	NE	NE						NE	NE	NE

NE – No exemption.
NE1 - Required for Protected Software.
NE2 – Required for Mission-essential Software.
NE3 – Required when software is developed for the classified environment by persons without a Q-clearance.
NE4 – Required when software will be installed in multiple environments.
SB – Required unless the SRB approves the exemption request.
PT – Required unless the Project Team approves the exemption request.

2.3.3.1 Functional Design Document

The Functional Design Document defines how the software product will be organizationally structured to satisfy the requirements identified in the Software Requirements Specifications. It is a detailed description of the configuration, components, interfaces, and data necessary before coding can begin. The functional security requirements are defined in the Functional Design Document. The functional design document may also include:

- design entities and dependencies,
- system input and output design,
- user interface design,
- system interface design,
- logical model, and
- data dictionary depending on the type and function of the software.

2.3.3.2 Stage Exit Position Response Form (SEPRF)

The SEPRF is the tool for obtaining approval of the deliverables and move forward into the next stage of development or implementation. Preliminary work may be performed on the next scheduled stage(s) per the Software Project Plan while reviews and approvals are being obtained to close the current stage officially. The roles and responsibilities for each stage identify the deliverables that each reviewer is approving or disapproving. An electronic form and a form template are available from SAM.

When an approved deliverable from a previous stage is modified in the current stage, such as updating the Software Project Plan, the SEPRF should note the approval of the updated deliverable. The designated approvers from the previous stage will approve the updates in the current stage.

2.3.3.3 Deliverable Approvals

Approvals of deliverables are based on the documents being reviewed and comments being resolved with responsible personnel prior to approval. Approval responsibility is assigned to the roles identified in Section 2.3.4. Approval of the stage deliverables is done via approval of the SEPRF.

2.3.3.4 Project Team Exemption Requests

For exemption requests that require disposition from the Project Team, the exemption rationale shall be documented. The documentation may be in the SPP, by an Exemption Request Form or in the comments field for the deliverable in SAM.

2.3.3.5 SRB Exemption Requests

For exemption requests that require disposition from the SRB, an Exemption Request Form or SAM Exemption Request must be submitted to the SRB. Exemption requests submitted to the SRB shall contain a statement of the deliverable that is being exempted and the rationale for the exemption. SRB exemption approvals/rejections shall be returned to the requester. The SRB minutes shall contain documentation of the exemption action.

2.3.4 Roles and Responsibilities in the Functional Design Stage

The Project Manager has the overall responsibility for the successful execution of the Functional Design Stage. The Y80 Procedure does not assign the completion of deliverables to specific Project Team members. The Project Manager assigns the completion of deliverables to the appropriate Project Team members. The roles and responsibilities listed below identify the role normally responsible for the action. Project Team members can be assigned multiple project roles by the Project Manager. Where multiple roles are listed as creating a deliverable, it is expected that these roles will work together to create the deliverable. Responsibilities other than approval responsibilities may be re-assigned to meet project needs.

2.3.4.1 System Owner

In the Functional Design Stage, the System Owner approves the Functional Design Document via the Stage Exit Position Response Form.

2.3.4.2 Project Manager

Responsibilities in the Functional Design Stage include:

- Coordinates system integration issues with other applications at Y-12. System integration will include user interface design, system interface design, and database development. Consistent design and development of application software will provide the foundation for a common data dictionary and ensure maximum enterprise operability for Y-12.
- Notify Project Managers of software that may be impacted as soon as possible when making changes to database design or the system interface.
- Identify any system integration incompatibilities as early as possible in the process and notify other Project Managers.
- Tracks and updates project progress against the Software Project Plan.
- Documents Project Team decisions regarding exemptions of Y80-102 deliverables.
- Submits a request for exemption via SAM, when feasible, or the Exemption Request Form.
- IF an Exemption Request Form is used for an exemption, THEN retains the approved or rejected exemption request in the project file.
- Approves the Functional Design Document via the Stage Exit Position Response Form.

2.3.4.3 Project Team

Responsibilities in the Functional Design Stage include:

- Concurs with PT deliverable exemptions.
- Concurs with SB deliverable exemption requests prior to the Project Manager submitting the request to the SRB.
- Responsible for any procurement activities as well as installation and testing.

2.3.4.4 User Point of Contact (POC)

In the Functional Design Stage, the User POC approves the Functional Design Document via the Stage Exit Position Response Form.

2.3.4.5 Computing and Telecommunications Security Organization (CTSO)

In the Functional Design Stage, CTSO reviews project deliverables for security implications at the discretion of the Cyber Security Program Manager.

2.3.4.6 CTSO Representative

Responsibilities in the Functional Design Stage for protected software include:

- Provides the functional security requirements for the software for the Functional Design Document.
- Approves the Functional Design Document via the Stage Exit Position Response Form.

2.3.4.7 Software Review Board (SRB)

Responsibilities in the Functional Design Stage include:

- Approves/rejects exemption requests for identified required deliverables.
- IF an Exemption Request form is used to request an exemption, THEN assigns an identification number to the exemption request.
- Reviews and approves or rejects requested exemptions, with rejection rationale.
- IF the Exemption Request form is used, THEN the exemption approval/rejection is marked, a copy is placed in the SRB files and the exemption request is returned to the Project Manager.

2.3.4.8 Lead Analyst

Responsibilities in the Functional Design Stage include:

- Creates the Functional Design Document.
- Approves the Functional Design Document via the Stage Exit Position Response Form.

2.4 System Design Stage

2.4.1 Purpose

To establish requirements and guidelines for documenting system design activities for proposed software systems for Y-12. The purpose of the System Design Stage is to translate the user-oriented functional design specifications into a set of technical, computer-oriented system design specifications and to design data structures and processes to the level of detail necessary to plan and execute the Programming through the Installation and Acceptance Stages.

2.4.2 Strategy

This section presents the Deliverables Matrix and the responsibilities for the System Design Stage. A graded approach based upon software source and software failure impact is used.

2.4.3 Stage Deliverables, Exemptions and Approvals

This section describes various deliverables that must be produced by the Project Team to fulfill the Y-12 SQA Program requirements. For deliverables that are form-based, the standard Y-12 form contained in SAM should be used when feasible. Templates for each deliverable are available from SAM and are used at the discretion of the Project Team. If the standard Y-12 form or template is not used by the Project Team, the team shall define the appropriate format, content, and level-of-detail that meet the project's needs and the intent of the deliverable. The DOE SEM contains additional guidance on selected deliverables and may be used as a reference by the Project Team. In all cases, the Project Team shall apply appropriate rigor in defining deliverable content, format, and level-of-detail, such that project quality is not compromised and unacceptable risk is not introduced into the project.

The Deliverables Matrix (Table 5) defines the graded approach based on the software source and software failure impact for implementing the System Design Stage at Y-12. Some deliverables are not exemptible and are marked "NE." Other deliverables are not exemptible under certain conditions and are marked "NEx" where "x" denotes the condition. When the software does not meet the specified condition, the deliverable is optional. Deliverables that may be exempted by the SRB provided adequate justification is provided are marked "SB." Deliverables that may be exempted by the Project Team provided adequate justification is documented are marked "PT." Optional deliverables are shaded in the table.

Responsibility for the approval of the specific deliverable is defined in Section 2.4.4.

Table 5 - Deliverables Matrix for the System Design Stage

Software Source	COTS				DEV INT				DA/GA				DEV EXT			
SFI	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
System Design Document						SB	SB	SB						SB	SB	SB
Conversion Plan		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1
Continuity of Operations Plan				NE2				NE2				NE2				NE2
Stage Exit Position Response Form		NE1	NE1	NE1		NE	NE	NE		NE1	NE1	NE1		NE	NE	NE

NE – No exemption.
NE1 - Required for Protected Software.
NE2 – Required for Mission-essential Software.
NE3 – Required when software is developed for the classified environment by persons without a Q-clearance.
NE4 – Required when software will be installed in multiple environments.
SB – Required unless the SRB approves the exemption request.
PT – Required unless the Project Team approves the exemption request.

2.4.3.1 System Design Document

The System Design Document is a translation of the requirements into a description of the software structure, software components, interfaces, physical model and data necessary to support the programming process.

2.4.3.2 Conversion Plan

If the software product replaces an existing automated or manual system, the Conversion Plan documents conversion procedures, outlines the installation of new and converted files and databases and converted manual records, coordinates the development of conversion programming, and plans the implementation of the conversion procedures. The plan must also define who will have access to the production information during the conversion including privileged users and what access controls will be in place.

Not all projects will require conversion of data. Documentation that no conversion is required for the project in the Software Project Plan or in the comments field in SAM fulfills this requirement in those cases.

2.4.3.3 Continuity of Operations Plan

The Continuity of Operations Plan documents the emergency response, backup operations, and post-disaster recovery planned for a mission-essential software product in the event of an emergency or disaster. The plan should cover total or partial cessation of operations, destruction or unavailability of data and files, and destruction or unavailability of equipment and facilities.

2.4.3.4 Stage Exit Position Response Form (SEPRF)

The SEPRF is the tool for obtaining approval of the deliverables to move forward into the next stage of development or implementation. Preliminary work may be performed on the next scheduled stage(s) if documented in the SPP while reviews and approvals are being obtained to close the current stage officially. The roles and responsibilities for each stage identify the deliverables that each reviewer is approving or disapproving. An electronic form and a form template are available from SAM.

When an approved deliverable from a previous stage is modified in the current stage, such as updating the Software Project Plan, the SEPRF should note the approval of the updated deliverable. The designated approvers from the previous stage will approve the updates in the current stage.

2.4.3.5 Deliverable Approvals

Approvals of deliverables are based on the documents being reviewed and comments being resolved with responsible personnel prior to approval. Approval responsibility is assigned to the roles identified in Section 2.4.4. Approval of the stage deliverables is done via approval of the SEPRF.

2.4.3.6 Project Team Exemption Requests

For exemption requests that require disposition from the Project Team, the exemption rationale shall be documented. The documentation may be in the SPP, by an Exemption Request Form or in the comments field for the deliverable in SAM.

2.4.3.7 SRB Exemption Requests

For exemption requests that require disposition from the SRB, an Exemption Request Form or SAM Exemption Request must be submitted to the SRB. Exemption requests submitted to the SRB shall contain a statement of the deliverable that is being exempted and the rationale for the exemption. SRB exemption approvals/rejections shall be returned to the requester. The SRB minutes shall contain documentation of the exemption action.

2.4.4 Roles and Responsibilities in the System Design Stage

The Project Manager has the overall responsibility for the successful execution of the System Design Stage. The Y80 Procedure does not assign the completion of deliverables to specific Project Team members. The Project Manager assigns the completion of deliverables to the appropriate Project Team members. The roles and responsibilities listed below identify the role normally responsible for the action. Project Team members can be assigned multiple project roles by the Project Manager. Where multiple roles are listed as creating a deliverable, it is expected that these roles will work together to create the deliverable. Responsibilities other than approval responsibilities may be re-assigned to meet project needs.

2.4.4.1 System Owner

In the System Design Stage, approves the following deliverables via the Stage Exit Position Response Form:

- Conversion Plan
- System Design Document
- Continuity of Operations Plan

2.4.4.2 Project Manager

Responsibilities in the System Design Stage include:

- Tracks and updates project progress against the Software Project Plan.
- Documents Project Team decisions regarding exemptions of Y80-102 deliverables.
- Submits a request for exemption via SAM, when feasible, or the Exemption Request Form.
- IF an Exemption Request Form is used for an exemption, THEN retains the approved or rejected exemption request in the project file.
- Approves the following deliverables via the Stage Exit Position Response Form:
    - Conversion Plan
    - System Design Document
    - Continuity of Operations Plan

2.4.4.3 Project Team

Responsibilities in the System Design Stage include:

- Concurs with PT deliverable exemptions.
- Concurs with SB deliverable exemption requests prior to the Project Manager submitting the request to the SRB.

2.4.4.4 User Point of Contact

Responsibilities in the System Design Stage include:

- Provides input for the Conversion Plan.
- Provides input for the System Design Document.
- Approves the following deliverables via the Stage Exit Position Response Form:
    - Conversion Plan
    - System Design Document
    - Continuity of Operations Plan

2.4.4.5 Computing and Telecommunications Security Organization (CTSO)

Responsibilities in the System Design Stage include:

- IF protected software and if production data will be loaded before acceptance of the security controls, THEN approves the Conversion Plan.
- IF mission-essential software, THEN approves the Continuity of Operations Plan via the Stage Exit Position Response Form.
- Reviews other project deliverables for security implications at the discretion of the Cyber Security Program Manager.

2.4.4.6 CTSO Representative

For protected software n the System Design Stage, the CTSO Representative approves the following deliverables via the Stage Exit Position Response Form:

- Conversion Plan
- Continuity of Operations Plan

2.4.4.7 Software Review Board (SRB)

Responsibilities in the System Design Stage include:

- Approves/rejects SB exemption requests for identified required deliverables.
- IF an Exemption Request form is used to request an exemption, THEN assigns an identification number to the exemption request.
- Reviews and approves or rejects requested exemptions, with rejection rationale.
- IF the Exemption Request form is used, THEN the exemption approval/rejection is marked, a copy is placed in the SRB files and the exemption request is returned to the Project Manager.

2.4.4.8 Lead Analyst

Responsibilities in the System Design Stage include:

- Creates the following deliverables:
  - Conversion Plan
  - System Design Document
  - Continuity of Operations Plan
- Approves the following deliverables via the Stage Exit Position Response Form:
  - System Design Document
  - Conversion Plan
  - Continuity of Operations Plan

2.4.4.9 Information System Security Officer (ISSO)

In the System Design Stage for mission-essential software residing in the classified environment, the ISSO approves the Continuity of Operations Plan via the Stage Exit Position Response Form.

2.5 Programming Stage

2.5.1 Purpose

To establish requirements for software code specifically developed or modified for use at Y-12 and testing requirements of security controls.

2.5.2 Strategy

This section presents the deliverables and responsibilities for the Programming Stage. A graded approach based upon software source and software failure impact is used.

2.5.3 Stage Deliverables, Exemptions and Approvals

This section describes various deliverables that must be produced by the Project Team to fulfill the Y-12 SQA Program requirements. For deliverables that are form-based, the standard Y-12 form contained in SAM should be used when feasible. Templates for each deliverable are available from SAM and are used at the discretion of the Project Team. If the standard Y-12 form or template is not used by the Project Team, the team shall define the appropriate format, content, and level-of-detail that meet the project's needs and the intent of the deliverable. The DOE SEM contains additional guidance on selected deliverables and may be used as a reference by the Project Team. In all cases, the Project Team shall apply appropriate rigor in defining deliverable content, format, and level-of-detail, such that project quality is not compromised and unacceptable risk is not introduced into the project.

The Deliverables Matrix (Table 6) defines the graded approach based on the software source and software failure impact for implementing the Programming Stage at Y-12. Some deliverables are not exemptible and are marked "NE." Other deliverables are not exemptible under certain conditions and are marked "NEx" where "x" denotes the condition. When the software does not meet the specified condition, the deliverable is optional. Deliverables that may be exempted by the SRB provided adequate justification is provided are marked "SB." Deliverables that may be exempted by the Project Team provided adequate justification is documented are marked "PT." Optional deliverables are shaded in the table.

Responsibility for the approval of the specific deliverable is defined in Section 2.5.4.

Table 6 - Deliverables Matrix for the Programming Stage

Software Source	COTS				DEV INT				DA/GA				DEV EXT			
SFI	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
Completed Units and Modules of Code					NE	NE	NE	NE					NE	NE	NE	NE
Integration and System Test Plan						PT	SB	NE						PT	SB	NE
Application Security Specification and Controls		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1
Application Certification Test Plan		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1
Continuity of Operations Test Plan				NE2				NE2				NE2				NE2
Code Review Documentation					NE3	NE3	NE3	NE3	NE3	NE3	NE3	NE3	NE3	NE3	NE3	NE3
Stage Exit Position Response Form		NE1	NE1	NE1		NE	NE	NE		NE1	NE1	NE1		NE	NE	NE

NE – No exemption.
NE1 - Required for Protected Software.
NE2 – Required for Mission-essential Software.
NE3 – Required when software is developed for the classified environment by persons without a Q-clearance.
NE4 – Required when software will be installed in multiple environments.
SB – Required unless the SRB approves the exemption request.
PT – Required unless the Project Team approves the exemption request.

2.5.3.1 Completed Units and Modules of Code

The source code and machine-readable code, databases, and other physical files needed to implement the software system.

2.5.3.2 Integration and System Test Plan

The Integration and System Test Plan documents the required testing necessary to verify the integrity of a module and its interfaces with other modules within the software structure. It is used to assure that the software product satisfies the project requirements, functions in the computer-operating environment, and successfully interfaces between operating procedures and other systems.

2.5.3.3 Application Security Specification and Controls (ASSC)

The ASSC is based on the security controls established in the Software Requirements Specification and/or the Functional Design Document. The ASSC deliverable details what security controls will be implemented and how the controls will be implemented. This deliverable is only required for protected software.

2.5.3.4 Application Certification Test Plan

The Application Certification Test Plan is a Y-12 security requirement. Security features of the application must be tested to assure proper operation. Specific tests are defined in the Application Certification Test Plan. This deliverable is only required for protected software.

2.5.3.5 Continuity of Operations Test Plan

The Continuity of Operations Test Plan is a Y-12 security requirement for mission-essential software. The test plan is to verify the recovery mechanisms and alternate work methods defined in the Continuity of Operations Plan are working properly. The scope and content of the Continuity of Operations test shall be defined by the Project Team with input from CTSO.

2.5.3.6 Code Review Documentation

The Code Review Documentation is a Y-12 security requirement. A Q-cleared Project Team member must review software developed for use in the classified environment by personnel that are not Q-cleared.

2.5.3.7 Stage Exit Position Response Form (SEPRF)

The SEPRF is the tool for obtaining approval of the deliverables and move forward into the next stage of development or implementation. Preliminary work may be performed on the next scheduled stage(s) if documented in the SPP while reviews and approvals are being obtained to close the current stage officially. The roles and responsibilities for each stage identify the deliverables that each reviewer is approving or disapproving. An electronic form and a form template are available from SAM.

When an approved deliverable from a previous stage is modified in the current stage, such as updating the Software Project Plan, the SEPRF should note the approval of the updated deliverable. The designated approvers from the previous stage will approve the updates in the current stage.

2.5.3.8 Deliverable Approvals

Approvals of deliverables are based on the documents being reviewed and comments being resolved with responsible personnel prior to approval. Approval responsibility is assigned to the roles identified in Section 2.5.4. Approval of the stage deliverables is done via approval of the SEPRF. The exception to this rule is the Continuity of Operations Test Report which is approved by signature on the deliverable.

2.5.3.9 Project Team Exemption Requests

For exemption requests that require disposition from the Project Team, the exemption rationale shall be documented. The documentation may be in the SPP, by an Exemption Request Form or in the comments field for the deliverable in SAM.

2.5.3.10 SRB Exemption Requests

For exemption requests that require disposition from the SRB, an Exemption Request Form or SAM Exemption Request must be submitted to the SRB. Exemption requests submitted to the SRB shall contain a statement of the deliverable that is being exempted and the rationale for the exemption. SRB exemption approvals/rejections shall be returned to the requester. The SRB minutes shall contain documentation of the exemption action.

2.5.4 Roles and Responsibilities for the Programming Stage

The Project Manager has the overall responsibility for the successful execution of the Programming Stage. The Y80 Procedure does not assign the completion of deliverables to specific Project Team members. The Project Manager assigns the completion of deliverables to the appropriate Project Team members. The roles and responsibilities listed below identify the role normally responsible for the action. Project Team members can be assigned multiple project roles by the Project Manager. Where multiple roles are listed as creating a deliverable, it is expected that these roles will work together to create the deliverable. Responsibilities other than approval responsibilities may be re-assigned to meet project needs.

2.5.4.1 System Owner

Responsibilities in the Programming Stage include:

- Provides requirements for the Application Security Specifications and Controls.
- Approves the Application Security Specifications and Controls and the Application Certification Test Plan via the Stage Exit Position Response Form.
- Approves the Continuity of Operations Test Plan via a signature on the document.

2.5.4.2 Project Manager

Responsibilities in the Programming Stage include:

- Tracks and updates project progress against the SPP.
- Documents Project Team decisions regarding exemptions of Y80-102 deliverables.
- Submits a request for exemption via SAM, when feasible, or the Exemption Request Form.
- IF an Exemption Request Form is used for an exemption, THEN retains the approved or rejected exemption request in the project file.
- Approves the following deliverables via the Stage Exit Position Response Form:
    - Completed Units and Modules of Code
    - Integration and System Test Plan
    - Application Security Specifications and Controls
    - Application Certification Test Plan
    - Code Review Documentation, when required
- Approves the Continuity of Operations Test Plan via signature on the document.

2.5.4.3 Project Team

Responsibilities in the Programming Stage include:

- Concurs with PT deliverable exemptions.
- Concurs with SB deliverable exemption requests prior to the Project Manager submitting the request to the SRB.

2.5.4.4 User Point of Contact

Responsibilities in the Programming Stage include:

- Participates in the creation of the Continuity of Operations Test Plan.
- Participates in the creation of the Application Certification Test Plan.

2.5.4.5 Computing and Telecommunications Security Organization (CTSO)

Responsibilities in the Programming Stage include:

- For mission-essential software approves the Continuity of Operations Test Plan via signature on the document.
- Reviews the code review documentation.
- Reviews other project deliverables for security implications at the discretion of the Cyber Security Program Manager.

2.5.4.6 CTSO Representative

Responsibilities in the Programming Stage include:

- Ensures that the appropriate mitigation strategies for documented risks are addressed in the ASSC.
- For protected software, approves the following deliverables via the Stage Exit Position Response Form:
    - Application Security Specifications and Controls
    - Application Certification Test Plan
    - Code Review Documentation

2.5.4.7 Software Review Board (SRB)

Responsibilities in the Programming Stage include:

- Approves/rejects exemption requests for identified required deliverables.
- IF an Exemption Request form is used to request an exemption, THEN assigns an identification number to the exemption request.
- Reviews and approves or rejects requested exemptions, with rejection rationale.
- IF the Exemption Request form is used, THEN the exemption approval/rejection is marked, a copy is placed in the SRB files and the exemption request is returned to the Project Manager.

2.5.4.8 Lead Analyst

Responsibilities in the Programming Stage include:

- Prepares the following deliverables:
  - Completed Units and Modules of Code
  - Integration and System Test Plan
  - Application Security Specifications and Controls
  - Application Certification Test Plan
  - Code Review Documentation, when required
- Approves the following deliverables via the Stage Exit Position Response Form:
  - Completed Units and Modules of Code
  - Integration and System Test Plan
  - Application Security Specifications and Controls
  - Application Certification Test Plan
  - Code Review Documentation, when required
- Approves the Continuity of Operations Test Plan via signature on the document.

2.6 Integration and Testing Stage

2.6.1 Purpose

To establish a requirement by which a developed or acquired software application system is integrated and tested for production use.

2.6.2 Strategy

A graded approach based upon software source and software failure impact is used for the Integration and Testing Stage.

Stress tests and other operational tests are performed in the Integration and Testing Stage as required.

2.6.3 Stage Deliverables, Exemptions and Approvals

This section describes various deliverables that must be produced by the Project Team to fulfill the Y-12 SQA Program requirements. For deliverables that are form-based, the standard Y-12 form contained in SAM should be used when feasible. Templates for each deliverable are available from SAM and are used at the discretion of the Project Team. If the standard Y-12 form or template is not used by the Project Team, the team shall define the appropriate format, content, and level-of-detail that meet the project's needs and the intent of the deliverable. The DOE SEM contains additional guidance on selected deliverables and may be used as a reference by the Project Team. In all cases, the Project Team shall apply appropriate rigor in defining deliverable content, format, and level-of-detail, such that project quality is not compromised and unacceptable risk is not introduced into the project.

The Deliverables Matrix (Table 7) defines the graded approach based on the software source and software failure impact for implementing the Integration and Testing Stage at Y-12. Some deliverables are not exemptible and are marked "NE." Other deliverables are not exemptible under certain conditions and are marked "NEx" where "x" denotes the condition. When the software does not meet the specified condition, the deliverable is optional. Deliverables that may be exempted by the SRB provided adequate justification is provided are marked "SB." Deliverables that may be exempted by the Project Team provided adequate justification is documented are marked "PT." Optional deliverables are shaded in the table.

Responsibility for the approval of the specific deliverable is defined in Section 2.6.4.

Table 7 - Deliverables Matrix for the Integration and Testing Stage

Software Source	COTS				DEV INT				DA/GA				DEV EXT			
SFI	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**
Acceptance Test Plan		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE
User's Manual		PT	NE	NE		PT	NE	NE		PT	NE	NE		PT	NE	NE
Training Plan			PT	SB			PT	SB			PT	SB			PT	SB
Installation Plan		PT	PT	NE4		PT	PT	NE4		PT	PT	NE4		PT	PT	NE4
Integration and System Test Materials and Report						PT	SB	NE						PT	SB	NE
Programmers' Reference Manual			PT	PT		PT	SB	SB			PT	PT		PT	SB	SB
Stage Exit Position Response Form		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE

NE – No exemption.
NE1 - Required for Protected Software.
NE2 – Required for Mission-essential Software.
NE3 – Required when software is developed for the classified environment by persons without a Q-clearance.
NE4 – Required when software will be installed in multiple environments.
SB – Required unless the SRB approves the exemption request.
PT – Required unless the Project Team approves the exemption request.

2.6.3.1 Acceptance Test Plan (ATP)

The ATP describes the tests to be performed to validate that the application meets specified requirements and satisfies the users' criteria for accepting the system. In cases where the new system is to replace an existing system, the test plan may include a period of parallel testing to ensure the integrity of the users' business operations. The ATP is generally performed by the end users of the application to ensure that the system functions as desired in their business environment.

2.6.3.2 User's Manual

The User's Manual provides the information users need to access, navigate through, and operate the software product. Typically described are product or component capabilities, limitations, options, permitted inputs, expected outputs, possible error messages, special instructions, and who to contact for help or additional information.

This documentation can be in printed form, electronic files, online documentation, or a quick reference card, accessed through the application or any combination.

2.6.3.3 Training Plan (TP)

The TP describes the training needed by users, the system owner, and the maintenance staff to implement and operate the system.

2.6.3.4 Installation Plan

The Installation Plan describes the tasks that must be performed to install the software in the production environment. The plan must address software issues like how to compile, link, and move the software into the production environment.

The plan must also address how to verify the software is operating correctly once installed and what documentation will be retained as evidence of the successful installation.

The plan may address relevant hardware, firmware, and communications equipment needed to operate the product in each environment.

2.6.3.5 Integration and System Test Materials and Report

The Integration and System Test Materials and Report result from the execution of the Integration and System Test Plan. The documentation from the test execution is retained to verify the integrity of a module and its interfaces with other modules within the software structure. The testing is used to verify that the software product satisfies the project requirements, functions in the computer-operating environment, and successfully interfaces between operating procedures and other systems.

2.6.3.6 Programmer's Reference Manual

The Programmer's Reference Manual contains programming information used by the maintenance staff to maintain the programs, databases, interfaces, and operating environment. It should provide the details necessary to implement modifications or enhancements.

2.6.3.7 Stage Exit Position Response Form (SEPRF)

The SEPRF is the tool for obtaining approval of the deliverables and move forward into the next stage of development or implementation. Preliminary work may be performed on the next scheduled stage(s) if documented in the SPP while reviews and approvals are being obtained to close the current stage officially. The roles and responsibilities for each stage identify the deliverables that each reviewer is approving or disapproving. An electronic form and a form template are available from SAM.

When an approved deliverable from a previous stage is modified in the current stage, such as updating the Software Project Plan, the SEPRF should note the approval of the updated deliverable. The designated approvers from the previous stage will approve the updates in the current stage.

2.6.3.8 Deliverable Approvals

Approvals of deliverables are based on the documents being reviewed and comments being resolved with responsible personnel prior to approval. Approval responsibility is assigned to the roles identified in Section 2.6.4. Approval of the stage deliverables is done via approval of the SEPRF.

2.6.3.9 Project Team Exemption Requests

For exemption requests that require disposition from the Project Team, the exemption rationale shall be documented. The documentation may be in the SPP, by an Exemption Request Form, or in the comments field for the deliverable in SAM.

2.6.3.10 SRB Exemption Requests

For exemption requests that require disposition from the SRB, an Exemption Request Form, or SAM Exemption Request must be submitted to the SRB. Exemption requests submitted to the SRB shall contain a statement of the deliverable that is being exempted and the rationale for the exemption. SRB exemption approvals/rejections shall be returned to the requester. The SRB minutes shall contain documentation of the exemption action.

2.6.4 Roles and Responsibilities in the Integration and Testing Stage

The Project Manager has the overall responsibility for the successful execution of the Integration and Testing Stage. The Y80 Procedure does not assign the completion of deliverables to specific Project Team members. The Project Manager assigns the completion of deliverables to the appropriate Project Team members. The roles and responsibilities listed below identify the role normally responsible for the action. Project Team members can be assigned multiple project roles by the Project Manager. Where multiple roles are listed as creating a deliverable, it is expected that these roles will work together to create the deliverable. Responsibilities other than approval responsibilities may be re-assigned to meet project needs.

2.6.4.1 System Owner

In the Integration and Testing Stage, responsibility includes approval of the following deliverables via the Stage Exit Position Response Form:

- User's Manual
- Training Plan
- Installation Plan
- Acceptance Test Plan

2.6.4.2 Project Manager

Responsibilities in the Integration and Testing Stage include:

- Coordinates the Integration and System Testing.
- Tracks and updates project progress against the SPP.
- Documents Project Team decisions regarding exemptions of Y80-102 deliverables.
- Submits a request for exemption via SAM, when feasible, or the Exemption Request Form.
- IF an Exemption Request Form is used for an exemption, THEN retains the approved or rejected exemption request in the project file.
- Approves the following deliverables via the Stage Exit Position Response Form:
  - Programmer's Reference Manual
  - Installation Plan
  - Integration and System Test Materials and Report
  - Acceptance Test Plan

2.6.4.3 Project Team

Responsibilities in the Integration and Testing Stage include:

- Concurs with PT deliverable exemptions.
- Concurs with SB deliverable exemption requests prior to the Project Manager submitting the request to the SRB.

2.6.4.4 User Point of Contact

Responsibilities in the Integration and Testing Stage include:

- Provides input to the User's Manual.
- Provides input to the Training Plan.
- Approves the following deliverables via the Stage Exit Position Response Form.
    - User's Manual
    - Training Plan
    - Installation Plan
    - Acceptance Test Plan

2.6.4.5 Computing and Telecommunications Security Organization (CTSO)

In the Integration and Testing Stage, CTSO reviews project deliverables for security implications at the discretion of the Cyber Security Program Manager.

2.6.4.6 CTSO Representative

For protected software, approves the Installation Plan via the Stage Exit Position Response Form.

2.6.4.7 Software Review Board (SRB)

Responsibilities in the Integration and Testing Stage include:

- Approves/rejects exemption requests for identified required deliverables.
- IF an Exemption Request form is used to request an exemption, THEN assigns an identification number to the exemption request.
- Reviews and approves or rejects requested exemptions, with rejection rationale.
- IF the Exemption Request form is used, THEN the exemption approval/rejection is marked, a copy is placed in the SRB files and the exemption request is returned to the Project Manager.

2.6.4.8 Lead Analyst

Responsibilities in the Integration and Testing Stage include:

- Participates in the Integration and System Testing.
- Creates the following deliverables:
  - User's Manual
  - Programmer's Reference Manual
  - Training Plan
  - Installation Plan
  - Acceptance Test Plan
  - Integration and System Test Materials and Report
- Approves the following deliverables via the Stage Exit Position Response Form:
  - User's Manual
  - Programmer's Reference Manual
  - Training Plan
  - Installation Plan
  - Acceptance Test Plan
  - Integration and System Test Materials and Report

2.6.4.9 Quality Engineer (QE)

In the Integration and Testing Stage, the QE approves the Acceptance Test Plan via the Stage Exit Position Response Form.

2.6.4.10 Product Engineer (PE)

For product software, the PE approves the Acceptance Test Plan via the Stage Exit Position Response Form.

2.7 Installation and Acceptance (I&A) Stage

2.7.1 Purpose

To establish a requirement by which a developed or acquired software application system is installed and acceptance tested for production use. The installation is on the production platform. Prior to beginning the acceptance testing, all software requirement deliverables from the previous stages must be approved by the Project Team.

2.7.2 Strategy

I&A Stage deliverables are based upon a graded approach that considers the impact of software failure and the software source. The completed Stage Exit Position Response Form for the I&A Stage is required before moving the software into service and beginning production use in the Maintenance Stage.

2.7.3 Stage Deliverables, Exemptions and Approvals

This section describes various deliverables that must be produced by the Project Team to fulfill the Y-12 SQA Program requirements. For deliverables that are form-based, the standard Y-12 form contained in SAM should be used when feasible. Templates for each deliverable are available from SAM and are used at the discretion of the Project Team. If the standard Y-12 form or template is not used by the Project Team, the team shall define the appropriate format, content, and level-of-detail that meet the project's needs and the intent of the deliverable. The DOE SEM contains additional guidance on selected deliverables and may be used as a reference by the Project Team. In all cases, the Project Team shall apply appropriate rigor in defining deliverable content, format, and level-of-detail, such that project quality is not compromised and unacceptable risk is not introduced into the project.

The Deliverables Matrix (Table 8) defines the graded approach based on the software source and software failure impact for implementing the Requirements Definition Stage at Y-12. Some deliverables are not exemptible and are marked "NE." Other deliverables are not exemptible under certain conditions and are marked "NEx" where "x" denotes the condition. When the software does not meet the specified condition, the deliverable is optional. Deliverables that may be exempted by the SRB provided adequate justification is provided are marked "SB." Deliverables that may be exempted by the Project Team provided adequate justification is documented are marked "PT." Optional deliverables are shaded in the table.

Responsibility for the approval of the specific deliverable is defined in Section 2.7.4.

.

Table 8 - Deliverables Matrix for the Installation and Acceptance Stage

Software Source	COTS				DEV INT				DA/GA				DEV EXT			
SFI	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**
Software Inventory (Revised)	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE
Operational Software Product	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE
Configuration Management Plan		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE
Acceptance Test Materials and Report		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE
Version Description Document		PT	PT	NE		PT	SB	NE		PT	PT	NE		PT	SB	NE
User Training Materials		PT	PT	SB		PT	PT	SB		PT	PT	SB		PT	PT	SB
Application Certification Test Materials and Report		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1
Continuity of Operations Test Materials and Report				NE2				NE2				NE2				NE2
Stage Exit Position Response Form		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE

NE – No exemption.
NE1 - Required for Protected Software.
NE2 – Required for Mission-essential Software.
NE3 – Required when software is developed for the classified environment by persons without a Q-clearance.
NE4 – Required when software will be installed in multiple operating system environments.
SB – Required unless the SRB approves the exemption request.
PT – Required unless the Project Team approves the exemption request.

2.7.3.1 Software Inventory (Revised)

The software inventory entry in SAM must reflect current information regarding a specific application. The information to be updated includes life cycle stage, Project Team roles (e.g., system owner, lead analyst), security certification, and any other items currently defined within the software inventory repository.

2.7.3.2 Operational Software Product

The Operational Software Product is software that satisfies the customer's requirements, has been accepted by the customer, has been deployed, and is ready for production use.

2.7.3.3 Configuration Management Plan

The Configuration Management Plan describes the project's processes for providing configuration management for both project software and documentation. The software configuration management portion of the plan details the method of providing software module maintenance and control, the process of performing a system build, and steps required to place a new release in production.

Depending on the system platform, such information as directories, files, symbols, command procedures, etc., involved in providing configuration management of software modules are listed and explained.

The plan must address whether emergency changes to software are allowed. An emergency change to software should parallel the standard approach to making software changes as closely as possible with most of the activities, including preparation of a Software Change Request (SCR) and modifications to documentation, being performed retroactively.

When the need for handling emergency changes to the software is required, the process describing how emergency software changes are to be initiated, who can initiate them, how they will be fully documented, and the time limit for approval must be defined in the Configuration Management Plan. Emergency change privileges shall be limited, and the emergency change shall be appropriately documented, reviewed, and approved after the emergency change was placed into operation.

The plan will identify the project documentation that shall be controlled and updated for each version release of the system and define the process by which document configuration management will be provided. Documents to be controlled may be any set of deliverables produced during the initial software development process including, but not limited to, the following:

- Software Project Plan
- Software Requirements Specifications
- Acceptance Test Plan
- Configuration Management Plan
- Programmer's Reference Manual
- Users' Manual
- User Training Materials

Certain security-related documents—the Applications Security Specifications and Controls (ASSC), the Protected Software Identification (PSI) form, and Risk Assessment—must be maintained on a continuing basis by the Project Team per Y-12 cyber security requirements. These documents need not be specified in the Configuration Management Plan.

2.7.3.4 Acceptance Test Materials & Report

Acceptance Tests at Y-12 will be done, where possible, in a QA or migration environment that closely duplicates the production environment. Acceptance Tests will be performed in the production environment only if necessary.

The Acceptance Test Materials and Report shall include the list of test participants, who performed the test, who verified the test results and determined what re-tests, if any, are required, and signature of the test participants documenting completion of the tests.

After the Acceptance Test is completed, an Operational Readiness Review will not be required before the software is placed in production. The System Owner's approval of the SEPRF serves as the System Owner's approval and acceptance of the software application system.

The Acceptance Test Materials & Report documents the formal testing usually conducted by the user's organization(s). It demonstrates the software's compliance with the system owner's requirements and acceptance criteria. It provides evidence that the test was completed, summarizes the test results, documents the decision whether to accept or reject the software system, and captures materials generated by execution of the Acceptance Test Plan. The Acceptance Test Materials & Report also documents any problems detected and corrected, and the project's plan for correcting any open problems reported. If the system is later enhanced or otherwise modified to implement approved SCRs, the system is again acceptance tested. The Maintenance Stage acceptance tests may focus on the system functions that were impacted by the code changes.

2.7.3.5 Version Description Document (VDD)

The VDD contains information pertinent to a specific release of the software system. It identifies the system, provides information about the current release (date, version number, release number), lists the individual SCR(s) included in the release, and briefly describes the functionality delivered in the version release.

The VDD also includes a list of application software components which make up a given release of the system, including modules, command files, data files, and other information as appropriate. This list can be either a complete list of all application software components that make up a specific release of the application or an incremental list of the software components that were modified for the specific current release. If the second approach is used, there must be a previously generated complete software component list available in the Project File, which would be generated no earlier than the first system release.

At the Project Team's discretion, the complete list may be generated in parallel with periodic significant releases, such as major upgrades, and then augmented via incremental lists for less significant releases.

The VDD may optionally include environment information specific to a release such as related hardware and software including the operating system, references to required versions of layered systems products, a list of documents altered per the Configuration Management Plan, and a list of processors on which the system was installed.

2.7.3.6 User Training Materials

Training may be provided by the Project Team, the user organization, or the Center for Continuing Education. Non-SQA requirements may dictate who is to provide the training, since Y-12 operating organizations typically have legally and contractually imposed training requirements. The Project Team is responsible for the preparation of user training material in conjunction with the training officer of the appropriate line organization.

User training materials are those materials created to train software users in the use of a software product. Examples of such materials include formal training manuals, "quick start" command summaries, and sample program outputs.

2.7.3.7 Application Certification Test Materials & Report

The Application Certification Test Materials & Report documents the testing of the application's security and access control features specified in the Application Security Specification and Control (ASSC) document. The testing is required for all protected software. The testing is conducted in accordance with the Application Certification Test Plan. Satisfactory completion of the tests insures that the security features of the application have been fully verified and are operable to specified standards. The Application Certification tests must be performed during the Installation and Acceptance Stage.

2.7.3.8 Continuity of Operations Test Materials and Report

A Continuity of Operations Test is required for mission-essential systems, regardless of software source. A Continuity of Operations Test must be performed prior to software acceptance and documented in a plan with accompanying materials and report. The Continuity of Operations Test verifies the alternate business measure and disaster recovery components are properly functioning.

2.7.3.9 Stage Exit Position Response Form (SEPRF)

The SEPRF is the tool for obtaining approval of the deliverables before moving forward into the Maintenance Stage. Production use of the software is not allowed until this SEPRF has been approved by all approvers. The roles and responsibilities for each stage identify the deliverables that each reviewer is approving or disapproving. An electronic form and a form template are available from SAM.

When an approved deliverable from a previous stage is modified in the current stage, such as updating the Software Project Plan, the SEPRF should note the approval of the updated deliverable. The designated approvers from the previous stage will approve the updates in the current stage.

2.7.3.10 Deliverable Approvals

Approvals of deliverables are based on the documents being reviewed and comments being resolved with responsible personnel prior to approval. Approval responsibility is assigned to the roles identified in Section 2.7.4. Approval of the deliverable is generally done via approval of the SEPRF.

2.7.3.11 Project Team Exemption Requests

For exemption requests that require disposition from the Project Team, the exemption rationale shall be documented. The documentation may be in the SPP, by an Exemption Request Form or in the comments field for the deliverable in SAM.

2.7.3.12 SRB Exemption Requests

For exemption requests that require disposition from the SRB, an Exemption Request Form or SAM Exemption Request must be submitted to the SRB. Exemption requests submitted to the SRB shall contain a statement of the deliverable that is being exempted and the rationale for the exemption. SRB exemption approvals/rejections shall be returned to the requester. The SRB minutes shall contain documentation of the exemption action.

2.7.4 Roles and Responsibilities for the Installation and Acceptance Stage

The Project Manager has the overall responsibility for the successful execution of the Installation and Acceptance Stage. The Y80 Procedure does not assign the completion of deliverables to specific Project Team members. The Project Manager assigns the completion of deliverables to the appropriate Project Team members. The roles and responsibilities listed below identify the role normally responsible for the action. Project Team members can be assigned multiple project roles by the Project Manager. Where multiple roles are listed as creating a deliverable, it is expected that these roles will work together to create the deliverable. Responsibilities other than approval responsibilities may be re-assigned to meet project needs.

2.7.4.1 Transfer of Responsibility

Typically, there is no formal transfer of responsibility from a Project Team to a Maintenance Team since members of the Project Team responsible for development of the software continue to support the software during the Maintenance Stage. As team members are lost, their responsibilities are reassigned to a new team member or are spread across the remaining team members.

2.7.4.2 System Owner

Responsibilities in the Installation and Acceptance Stage include:

- Updates the software inventory and deliverables information in SAM.
- Approves the following deliverables via the Stage Exit Position Response Form:
    - User Training Materials
    - Acceptance Test Materials and Report
    - Operational Software Product
    - Application Certification Test Materials and Report
    - Configuration Management Plan
    - Version Description Document
- Approves the Continuity of Operations Test Materials and Report via a signature on the document.

2.7.4.3 Project Manager

Responsibilities in the Installation and Acceptance Stage include:

- Tracks and updates project progress against the Software Project Plan.
- Documents Project Team decisions regarding exemptions of Y80-102 deliverables.
- Prepares the Configuration Management Plan.
- Transfers the Project File to the System Owner after software acceptance.
- Submits a request for exemption via SAM, when feasible, or the Exemption Request Form.
- IF an Exemption Request Form is used for an exemption, THEN retains the approved or rejected exemption request in the project file.
- Approves the following documents via the Stage Exit Position Response Form:
    - Acceptance Test Materials and Report
    - Operational Software Product
    - Application Certification Test Materials and Report
    - Configuration Management Plan
- Approves the Continuity of Operations Test Materials and Report via a signature on the document.

2.7.4.4 Project Team

Responsibilities in the Installation and Acceptance Stage include:

- Concurs with PT deliverable exemptions.
- Concurs with SB deliverable exemption requests prior to the Project Manager submitting the request to the SRB.

2.7.4.5 User Point of Contact (POC)

Responsibilities in the Installation and Acceptance Stage include:

- Prepares the User Training Materials.
- Approves the following deliverables via the Stage Exit Position Response Form:
    - Acceptance Test, Materials and Report
    - Operational Software Product
    - User Training Materials

2.7.4.6 Computing and Telecommunications Security Organization (CTSO)

Responsibilities in the Installation and Acceptance Stage include:

- IF mission-essential software, THEN approves the Continuity of Operations Test Materials and Report via a signature on the document.
- Reviews other project deliverables for security implications at the discretion of the Cyber Security Program Manager.

2.7.4.7 CTSO Representative

For protected software, responsibilities in the Installation and Acceptance Stage include:

- Approves the following deliverables via the Stage Exit Position Response Form.
  - Application Certification Test Material and Report
  - Configuration Management Plan
- Approves the Continuity of Operations Test Materials and Report via a signature on the document.

2.7.4.8 Software Review Board (SRB)

Responsibilities in the Installation and Acceptance Stage include:

- Approves/rejects exemption requests for identified required deliverables.
- IF an Exemption Request form is used to request an exemption, THEN assigns an identification number to the exemption request.
- Reviews and approves or rejects requested exemptions, with rejection rationale.
- IF the Exemption Request form is used, THEN the exemption approval/rejection is marked, a copy is placed in the SRB files and the exemption request is returned to the Project Manager.

2.7.4.9 Lead Analyst

Responsibilities in the Installation and Acceptance Stage include:

- Assures the Operational Software Product is installed and ready for use.
- Participates in Acceptance Testing.
- Prepares the following deliverables:
    - Version Description Document
    - Operational Software Product
    - Acceptance Test Materials and Report
    - Application Certification Test Materials and Report
    - Continuity of Operations Test Materials and Report
- Assigns the Version Identification number to the software.
- Approves the following deliverables via the Stage Exit Position Response Form:
    - Configuration Management Plan
    - User Training Materials
    - Version Description Document
    - Operational Software Product
    - Acceptance Test Materials and Report
    - Application Certification Test Materials and Report
- Approves the Continuity of Operations Test Materials and Report via a signature on the document.

2.7.4.10 Quality Engineer (QE)

Responsibilities in the Installation and Acceptance Stage include:

- Participates in Acceptance Testing.
- Verifies the Software Inventory in SAM has been updated and reflects the current software status.
- Performs the quality assurance review.
- Approves the following deliverables via the Stage Exit Position Response Form:
    - Acceptance Test Materials and Report
    - Configuration Management Plan
    - Version Description Document

2.7.4.11 Product Engineer (PE)

For product software, the PE responsibilities in the Installation and Acceptance Stage include:

- Participates in Acceptance Testing.
- Approves the following deliverables via the Stage Exit Position Response Form:
    - Acceptance Test Materials and Report
    - Version Description Document

2.7.4.12 Information System Security Officer (ISSO)

In the Installation and Acceptance Stage for mission-essential software that resides on a classified system, the ISSO approves the Continuity of Operations Test Materials and Report via a signature on the document.

2.8 Maintenance Stage

2.8.1 Purpose

To establish how maintenance and enhancement of software at Y-12 is managed and controlled to ensure appropriate quality, project management, and technical implementation of the process.

2.8.2 Strategy

The Maintenance Stage begins upon the initial deployment of the software and continues until the software is retired. This stage will occur repetitively based on the number of changes required for the system. During this stage, the Project Team should continue to use an appropriate life cycle approach for changes to ensure continued quality. Given the broad range of changes that could occur, the Project Team should use appropriate judgment in deciding whether to have a full project life cycle or a greatly reduced one per the scope of this instruction for each change. The Project Team's decisions should be compliant with the approved Configuration Management Plan.

Changes or revisions to software in the Maintenance life cycle will be evaluated in a manner consistent with the method described in this section of the instruction. A repeat of the entire Evaluation Process is not required for each approved software change, but the Project Team has the responsibility to ensure that the constraints put in place by the original evaluation of SFI, protected software, and product determination are not violated during the Maintenance Stage.

2.8.2.1 Iterative Process

The Maintenance Stage is repetitive until the application is retired from use. In general, the sequence of steps follows a pattern similar to the following:

1. The application runs steady state until a problem is discovered or a change is needed.

2. A SCR is submitted and approved to authorize a needed change or problem fix. If the change is large and complex, in the view of the Project Team, then a Software Project Plan may be prepared.

3. The Project Team considers impact on the current SFI, product, and protected designations.

4. The Project Team updates documents described in the Configuration Management Plan for the system.

5. The Project Team makes the change(s) authorized by one or more SCRs.

6. The Project Team performs Integration/System and Acceptance Testing. Installation Testing is performed if needed.

7. A Version Description Document (VDD) is prepared.

8. A Stage Exit Position Response Form is signed to indicate acceptance of the changes made.

9. Data conversion from the old version to the new version of the software is performed if necessary.

10. The new software is installed.

2.8.2.2 Independent Activities

Independently of these activities, there are other steps that may be performed in certain situations such as:

- The software inventory (SAM) is updated if operating parameters change to warrant a modification or re-evaluation by the Software Review Board (SRB).
- A Data Change Request (DCR) is prepared for data changes as needed, per guidelines in this instruction (no SCR, VDD, or SEPRF is needed for a DCR).
- Continuity of Operations Testing is performed for mission-essential systems on a yearly basis.
- A code review is performed if developers without a Q-clearance modify the software prior to installation of the software on the classified processor.
- An Application Certification Test must be performed every three years or when security features change.

2.8.2.3 Emergency Change Process

When the need for handling emergency changes to the software is required, the process describing how emergency software changes are to be initiated, who can initiate them, how they will be fully documented, and the time limit for approval must be defined in the Configuration Management Plan. Emergency change privileges shall be limited, and the emergency change shall be appropriately documented, reviewed, and approved after the emergency change was placed into operation.

2.8.2.4 System Interfaces

The Project Manager and the Lead Analyst must remain aware of the potential for modifications to affect other systems that interface with the software. The possible effects of modifications should be communicated as early as possible to owners of related systems and should be factored into the overall process of planning for modifications and to any related evaluation of the system that occurs over the entire project life cycle.

2.8.2.5 Software Evaluation

Modifications made during the Maintenance Stage could change the approved SFI, protected status, and product determination, resulting in the need for more or less security or QA oversight. It is the responsibility of the Project Team, and the Project Manager in particular, to ensure that any modifications made during maintenance are accounted for with respect to the software's overall characterization in relation to these evaluation factors.

It is not necessary to repeat the entire software evaluation process for each SCR processed during the Maintenance Stage. In general, the Project Team should maintain overall rigor during the Maintenance Stage that was associated with the failure impact and other factors as determined during the original project evaluation. It is recognized that the possible failure impact or other factors associated with a given SCR can vary widely. Therefore, when evaluating a given SCR for processing, the Project Team has the responsibility to properly characterize the failure impact associated with a particular change and apply appropriate management and control of the implementation of that same change as defined within this instruction. In some cases, this will permit a given SCR to be processed at a failure impact that is different from that of the entire system with respect to documentation or approvals required. An example might be the changing of header text on a screen or report for SFI 3 software, which could be treated as a minor change that is of a failure impact level 1 for purposes of the given Maintenance Stage iteration. The System Owner and Project Manager have ultimate responsibility for making sure that the system's overall failure impact constraints are not violated during the Maintenance Stage while processing individual SCRs.

2.8.3 Stage Deliverables, Exemptions and Approvals

The large variations in software maintenance scope and project type make it difficult to define an exact set of deliverables for all Maintenance Stage iterations across all projects. Therefore, the Deliverable Matrix lists those items, which are considered essential for overall software quality during a generic Maintenance Stage. As with the other sections of this instruction, the Project Team has significant flexibility to add other deliverables that may help to ensure software quality in a particular situation, including deliverables not required by this instruction. In all cases, the Project Team shall apply appropriate rigor in defining deliverable content, format, and level-of-detail, such that project quality is not compromised and unacceptable risk is not introduced into the project or software.

The Deliverables Matrix (Table 9) defines the graded approach based on the software source and software failure impact for implementing the Maintenance Stage at Y-12. Some deliverables are not exemptible and are marked "NE." Other deliverables are not exemptible under certain conditions and are marked "NEx" where "x" denotes the condition. When the software does not meet the specified condition, the deliverable is optional. Deliverables that may be exempted by the SRB provided adequate justification is provided are marked "SB." Deliverables that may be exempted by the Project Team provided adequate justification is documented are marked "PT." Optional deliverables are shaded in the table.

Responsibility for the approval of the specific deliverable is defined in Section 2.8.4.

Table 9 - Deliverables Matrix for the Maintenance Stage

Software Source	COTS				DEV INT				DA/GA				DEV EXT			
SFI	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
Software Inventory (Revised)	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE
Software Change Request (SCR)		PT	SB	SB		PT	SB	NE		PT	SB	SB		PT	SB	NE
Version Description Document		PT	SB	NE		PT	SB	NE		PT	SB	NE		PT	SB	NE
Operational Software Product	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE	NE
Configuration Management Plan		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE
Software Project Plan		PT	PT	PT		PT	PT	PT		PT	PT	PT		PT	PT	PT
Integration and System Test Plan, Materials and Report						PT	SB	NE						PT	SB	NE
Acceptance Test Plan, Materials and Report		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE
Risk Assessment		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1
Application Security Specification and Controls		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1
Application Certification Test Plan, Materials and Report		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1		NE1	NE1	NE1
Continuity of Operations Test Plan, Materials and Report				NE2				NE2				NE2				NE2
Code Review Documentation					NE3	NE3	NE3	NE3	NE3	NE3	NE3	NE3	NE3	NE3	NE3	NE3
Data Change Request		NE5	NE5	NE5		NE5	NE5	NE5		NE5	NE5	NE5		NE5	NE5	NE5
Stage Exit Position Response Form		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE

NE – No exemption.
NE1 – Required for protected software.
NE2 – Required for mission-essential software.
NE3 – Required when software is developed for the classified environment by persons without a Q-clearance.
NE4 – Required when software will be installed in multiple environments.
NE5 – See deliverable definition for when required.
SB – Required unless the SRB approves the exemption request.
PT – Required unless the Project Team approves the exemption request.

2.8.3.1 Software Change Request (SCR)

The SCR documents the basic request for a software application change and provides evidence of agreement to proceed with the change. Unlike some other deliverables, the SCR form is signed before work proceeds on modifying an application, as opposed to at the exit point of Maintenance Stage iteration. This is because the SCR serves as agreement to perform a change to the application. Therefore, it must be approved prior to beginning work on a given modification. One or more approved SCR forms serve as the initiator for a given iteration of the Maintenance Stage.

2.8.3.2 Version Description Document (VDD)

The VDD includes a list of the components that are included in a software version release and an optional list of processors where the software is installed. The VDD may also include pertinent information about the operating environment, such as the version numbers of associated products, as deemed appropriate by the Project Team to describe the entire operating environment for a release.

2.8.3.3 Operational Software Product

The Operational Software Product is software that satisfies the customer's requirements, has been accepted by the customer, has been deployed, and is ready for production use. Over the life of a software system, the Maintenance Stage will likely deliver multiple versions of the Operational Software Product as the system's functionality is enhanced, the software environment (e.g., operating system, database management system) changes, etc.

2.8.3.4 Configuration Management Plan (CMP) and Documentation Revisions

The Project Team defined the documents and related items to be maintained for the Maintenance Stage in the Configuration Management Plan (CMP) during the Installation and Acceptance Stage. During a given iteration of the Maintenance Stage, anywhere from none to all of these documents may need to be revised and updated. It is up to the Project Team to ensure that appropriate updates are made to the items defined in the CMP.

2.8.3.5 Software Inventory (Revised)

The software inventory entry in SAM must reflect current information regarding a specific application. The information to be updated includes life cycle stage, Project Team roles (e.g., system owner, lead analyst), security certification, and any other items currently defined within the software inventory repository.

2.8.3.6 Software Project Plan (SPP)

It is the Project Manager's responsibility to determine when the magnitude of a set of software changes warrants the development of a SPP to ensure project success. For small software changes, whether individual or grouped, such as "bug fixes," a SPP may not be required, since most pertinent information is included on the SCR form and other accompanying documentation. For changes of significant magnitude, such as the addition or change of major application features, functionality, and security controls; porting of code to other environments; and other such related changes, it is strongly recommended that a SPP be prepared in advance of the initiation of work on the system.

2.8.3.7 Integration and System Test Plan, Materials and Report

Integration and system testing activities will be done as needed during the software change testing process. The Integration and System Test Plan, Materials and Report documents the required testing and results.

With agreement between the Project Manager and the System Owner, the system and acceptance testing may be merged, if the intent of the two testing components is met. Both testing regimens are important in ensuring software quality, but specific Maintenance Stage iterations may achieve the desired results via a combined set of tests.

2.8.3.8 Acceptance Test Plan, Materials and Report

Acceptance testing activities will be done to validate that the software changes meet the specified requirements, satisfies the users' criteria for accepting the software, and assure no side effects were introduced as a result of the software changes. The Acceptance Test Plan, Materials and Report deliverable documents the required testing and results.

With agreement between the Project Manager and the System Owner, the system test and acceptance test may be merged, if the intent of the two testing components is met. Both testing regimens are important in ensuring software quality, but specific maintenance stage iterations may achieve the desired results via a combined set of tests.

2.8.3.9 Risk Assessment

The Risk Assessment should be reviewed and updated as necessary when security significant changes are made to protected applications. The Risk Assessment is also reviewed and updated as necessary during re-certification of the software or at the direction of CTSO.

2.8.3.10 Application Security Specification and Controls (ASSC)

The ASSC is updated as necessary when security significant changes are made to the software. The ASSC must be reviewed and updated at a minimum of every three years.

2.8.3.11 Continuity of Operations Test Plan, Materials and Report

A Continuity of Operations Test is required for mission-essential systems, regardless of software source. A Continuity of Operations Test must be performed on at least a yearly basis and documented in a plan with accompanying materials and report. If there are significant hardware or software changes to the system that impact operations, the test may be performed earlier than one year at the discretion of the System Owner. Each performance of the test serves to fulfill the yearly test requirement.

2.8.3.12 Application Certification Test Plan, Materials and Report

An Application Certification Test is required for Protected Software. This test must be performed a minimum of every three years and documented in a plan with accompanying materials and report. If there are significant security changes to the application prior to the end of the three-year period, the test should be performed earlier to ensure compliance. A given iteration of the test may cover only specified security changes rather than an entire application test, if agreed to by CTSO and the System Owner.

2.8.3.13 Code Review Documentation

The Code Review Documentation is a Y-12 security requirement. A Q-cleared Project Team member must review software developed for use in the classified environment by personnel that are not Q-cleared.

2.8.3.14 Data Change Request (DCR)

The DCR documents a formal request for a data modification that cannot be accomplished via the standard application user interface. Typically, changes that require a DCR are accomplished via macros, queries, or related methods. These changes often involve mass modifications to a set of data, but may be as simple as adding additional database entries to an existing table. A DCR does not require a Maintenance Stage exit iteration or issuance of a Version Description Document (VDD).

A DCR is only to be filled out for data changes that are separate from a SCR driven application modification. Therefore, if a SCR includes a data modification as part of the overall application change, a DCR is not required. However, if no automated audit trail is in use for a data change; there is no defined and approved alternate data update method in the Configuration Management Plan; and the change is one of the following types, then a DCR must be prepared for all software sources:

- classified data is modified;
- a data change which is a level 2 or 3 impact level (e.g., change of parameters on a life safety system);
- a data change which has possible waste, fraud, or abuse implications (e.g., financial data, personnel records, security records) as defined by the system owner.

If an automated audit trail is in use for a change of the above type, then a DCR is optional. In all other cases, a DCR is fully optional. Data changes on a system that is rated at failure impact level 2 or 3 are not automatically rated at that same level, but should be considered on a case by case basis for individual impact.

The DCR is approved at time of issuance not via the SEPRF.

2.8.3.15 Stage Exit Position Response Form (SEPRF)

The SEPRF is the tool for obtaining approval of the deliverables in accordance with the Configuration Management Plan. The roles and responsibilities for each the original stage where the deliverable was approved identify the deliverables that each reviewer is approving or disapproving. In addition, the SEPRF documents the approval of the all revisions, testing, and other tasks and items associated with that release. An electronic form and a form template are available from SAM.

A Stage Exit Position Response Form must be completed for each new release of an application during the Software Maintenance Stage. The signatures associated with the SEPRF indicate the approvals of project stakeholders for all revisions, testing, and other tasks and items associated with that release.

2.8.3.16 Deliverable Approvals

Approvals of deliverables are based on the documents being reviewed and comments being resolved with responsible personnel prior to approval. Approval responsibility is assigned to the roles identified in Section 2.8.4. Approval of the deliverable is generally done via approval of the SEPRF. Exceptions to this rule are the Risk Assessment, Application Certification Test Report and the Continuity of Operations Test Report, which must be prepared and approved periodically regardless of the need for software releases. These exceptions are due to the independent status of these documents.

2.8.4 Roles and Responsibilities in the Maintenance Stage

The Project Manager has the overall responsibility for the successful execution of the Maintenance Stage. This procedure does not assign the completion of deliverables to specific Project Team members. The Project Manager assigns the completion of deliverables to the appropriate Project Team members. The roles and responsibilities listed below identify the role normally responsible for the action. Where multiple roles are listed as creating a deliverable, it is expected that these roles will work together to create the deliverable. Responsibilities other than approval responsibilities may be re-assigned to meet project needs.

2.8.4.1 Transfer of Responsibility

During the Maintenance Stage as team members leave the project, their responsibilities are reassigned to a new team member or are spread across the remaining team members.

2.8.4.2 System Owner

Responsibilities in the Maintenance Stage include:

- Works with the appropriate funding source to provide funding for continued support of the software.
- Reviews and updates the Software Inventory and Deliverable information in SAM at least annually.
- Assures the Project File is maintained.
- Accepts software for production use via approval on the Stage Exit Position Response Form.
- Reviews emergency changes to ensure that they are appropriate and that the required documentation has been completed.
- IF mission-essential software, THEN defines the scope and content of the Continuity of Operations test with input from CTSO.
- Approves the Software Change Request at the time of request.
- Approves the Data Change Request at the time of request.
- Approves the Continuity of Operations Test Plan, Materials and Report via signature on the document.
- Approves revisions to the documentation via the Stage Exit Position Response Form:
    - Software Project Plan
    - Version Description Document
    - Risk Assessment
    - Application Security Specification and Controls
    - Operational Software Product
    - Acceptance Test Plan, Materials and Report
    - Configuration Management Plan
    - Application Certification Test Plan ,Materials and Report
- Submits a request for exemption via SAM, when feasible, or the Exemption Request Form.
- IF an Exemption Request Form is used for an exemption, THEN retains the approved or rejected exemption request in the project file.

2.8.4.3 Project Manager

Responsibilities in the Maintenance Stage include:

- Creates the Software Project Plan or updates the existing SPP as appropriate for phase.
- Performs evaluation of system failure impact, protected or mission essentiality, or other system characteristics if warranted by modifications.
- Updates the Software Inventory in SAM.
- Approves revisions to the documentation via a Stage Exit Position Response Form:
    - Software Project Plan
    - Configuration Management Plan
    - Acceptance Test Plan, Materials and Report
    - Integration and System Test Plan, Materials and Report
    - Application Certification Test Plan, Materials and Report
    - Risk Assessment
    - Application Security Specification and Controls
    - Operational Software Product
    - Version Description Document
- Approves the Continuity of Operations Test Plan, Materials and Report via signature on the document.

2.8.4.4 Project Team

Responsibilities in the Maintenance Stage include:

- Concurs with PT deliverable exemptions.
- Concurs with SB deliverable exemption requests prior to the Project Manager submitting the request to the SRB.

2.8.4.5 User Point of Contact

Responsibilities in the Maintenance Stage include:

- Participates in the User Acceptance Test.
- Approves the following deliverables via a Stage Exit Position Response Form:
    - Configuration Management Plan
    - Software Project Plan
    - Acceptance Test Plan Materials and Report
    - Operational Software Product

2.8.4.6 Computing and Telecommunications Security Organization (CTSO)

Responsibilities in the Maintenance Stage include:

- Reviews any project deliverables for security implications at the discretion of the Cyber Security Program Manager.
- IF mission-essential software, THEN –
    - Assist the System Owner in defining the scope and content of the Continuity of Operations test
    - Approves the Continuity of Operations Test Plan, Materials and Report via a signature on the document.
- Approves the Risk Assessment via the Stage Exit Position Response Form.

2.8.4.7 CTSO Representative

For protected software, responsibilities in the Maintenance Stage include:

- Determines if the proposed change is security significant.
- Specifies the time period to document emergency changes that affect the software's security features.
- Approves the following deliverables via a Stage Exit Position Response Form.
    - Application Certification Test Plan, Materials and Report
    - Software Change Request when the change is security significant
    - Code Review Documentation
    - Application Security Specifications and Controls
    - Configuration Management Plan
    - Software Project Plan

2.8.4.8 Lead Analyst

Responsibilities in the Maintenance Stage include:

- Assigns the Version Identification number to the software and prepares the Version Description Document.
- Performs software modifications and assures the software is modified to the meet the requirements specified in the SCR.
- Assures the Operational Software Product is installed and ready for use.
- Performs the acceptance, application certification, installation, and integration and system testing as necessary.
- Performs the code review as necessary.
- Revises the Configuration Management Plan and other software documentation per the Configuration Management Plan.
- Completes emergency changes when authorized.
- Prepares and approves the Software Change Request and the Data Change Request as requests are made.
- Prepares the following deliverables:
    - Acceptance Test Plan, Materials and Report
    - Application Certification Test Plan, Materials and Report
    - Continuity of Operations Test Plan, Materials and Report
    - Integration and System Test Plan, Materials and Report
- Approves the Continuity of Operations Test Plan, Materials and Report via signature on the document.
- Approves the following deliverables via a Stage Exit Position Response Form:
    - Integration and System Test Materials and Report
    - Software Change Request
    - Data Change Request
    - Version Description Document
    - Operational Software Product
    - Code Review Documentation
    - Configuration Management Plan
    - Software Project Plan
    - Application Certification Test Plan, Materials and Report
    - Acceptance Test Plan, Materials and Report

2.8.4.9 Software Review Board (SRB)

Responsibilities in the Maintenance Stage include:

- Approves/rejects exemption requests for identified required deliverables.
- IF an Exemption Request form is used to request an exemption, THEN assigns an identification number to the exemption request.
- Reviews and approves or rejects requested exemptions, with rejection rationale.
- IF the Exemption Request form is used, THEN the exemption approval/rejection is marked, a copy is placed in the SRB files and the exemption request is returned to the Project Manager.

2.8.4.10 Quality Engineer (QE)

Responsibilities in the Maintenance Stage include:

- IF product software, THEN participates in Acceptance Testing.
- Approves the following deliverables via a Stage Exit Position Response Form:
    - Acceptance Test Plan, Materials and Report
    - Configuration Management Plan
    - Version Description Document
    - Software Project Plan

2.8.4.11 Product Engineer (PE)

For product software, responsibilities in the Maintenance Stage include:

- Participates in Acceptance Testing.
- Approves the following deliverables via a Stage Exit Position Response Form:
    - Acceptance Test Plan, Materials and Report
    - Version Description Document
    - Software Project Plan

2.8.4.12 Information System Security Officer (ISSO)

When the software resides on a classified system, responsibilities in the Maintenance Stage include:

- Approves the Application Certification Test Plan, Materials and Report via the Stage Exit Position Response Form.
- IF mission-essential software on a classified system, THEN approves the Continuity of Operations Test Plan, Materials and Report via signature on the document.

2.9 Retirement Stage

2.9.1 Purpose

To establish the requirements by which a developed or acquired software application system is retired from production use. Retirement could result from various situations including replacement by a new system, lack of business need, or loss of funding.

2.9.2 Strategy

Disposition of the software is dependent upon retirement conditions. Record retention requirements, preservation of software and documentation, and removal of all software and accounts from the production environment must be addressed in the retirement planning. All software, documentation, and data must comply with Procedure Y15-101, "Records Management."

When data is not being converted to a new system, record retention is a prime concern. Software to manipulate the data must be preserved for the same period of time the data is retained, to ensure that the data may be retrieved for administrative, legal, or fiscal purposes.

When data is being converted to a new system, record-keeping requirements will be minimal. The replacement system must comply with the records management requirements.

Work with the records analysts or personnel responsible for Y-12 Records Management to establish appropriate action for long-term storage, scheduling, and disposition of the software, project documentation, and data.

For the Retirement Stage, the focus becomes data preservation and not software failure impact. Key areas to address as the software is retired are:

- The software and documentation are properly scheduled and preserved for record retention requirements and retrieval.
- If the data has future value or record retention requirements, it is properly preserved with an acceptable method to access.
- The software and associated files have been properly removed from the operating environment and access removed.

2.9.3 Stage Deliverables, Exemptions and Approvals

This section describes various deliverables that must be produced by the Project Team to fulfill the Y-12 SQA Program requirements. For deliverables that are form-based, the standard Y-12 form contained in SAM should be used when feasible. Templates for each deliverable are available from SAM and are used at the discretion of the Project Team. If the standard Y-12 form or template is not used by the Project Team, the team shall define the appropriate format, content, and level-of-detail that meet the project's needs and the intent of the deliverable. The DOE SEM contains additional guidance on selected deliverables and may be used as a reference by the Project Team. In all cases, the Project Team shall apply appropriate rigor in defining deliverable content, format, and level-of-detail, such that project quality is not compromised and unacceptable risk is not introduced into the project.

The Deliverables Matrix (Table 10) defines the graded approach based on the software source and software failure impact for implementing the Retirement Stage at Y-12. Some deliverables are not exemptible and are marked "NE." Other deliverables are not exemptible under certain conditions and are marked "NEx" where "x" denotes the condition. When the software does not meet the specified condition, the deliverable is optional. Deliverables that may be exempted by the SRB provided adequate justification is provided are marked "SB." Deliverables that may be exempted by the Project Team provided adequate justification is documented are marked "PT." Optional deliverables are shaded in the table.

Responsibility for the approval of the specific deliverable is defined in Section 2.9.4.

Table 10 - Deliverables Matrix for the Retirement Stage

Software Source	COTS				DEV INT				DA/GA				DEV EXT			
SFI	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**
Software Inventory (Revised)		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE
Software Retirement Checklist		NE	NE	NE		NE	NE	NE		NE	NE	NE		NE	NE	NE
Data Change Request		NE5	NE5	NE5		NE5	NE5	NE5		NE5	NE5	NE5		NE5	NE5	NE5
Software Retirement Plan		PT	SB	SB		PT	SB	SB		PT	SB	SB		PT	SB	SB

NE – No exemption.
NE1 - Required for Protected Software.
NE2 – Required for Mission-essential Software.
NE3 – Required when software is developed for the classified environment by persons without a Q-clearance.
NE4 – Required when software will be installed in multiple environments.
NE5 – See deliverable definition for when required.
SB – Required unless the SRB approves the exemption request.
PT – Required unless the Project Team approves the exemption request.

2.9.3.1 Software Inventory (Revised)

The software inventory entry in SAM must reflect current information regarding a specific application. The information to be updated includes life cycle stage, Project Team roles (e.g., system owner, lead analyst), security certification, and any other items currently defined within the software inventory repository.

2.9.3.2 Software Retirement Checklist

The Software Retirement Checklist provides common activities that may be required to retire the software. This checklist will be used by the Project Team to determine and document the necessary retirement needs. This checklist provides the input to the Software Retirement Plan and can be incorporated in the Software Retirement Plan at the discretion of the System Owner.

2.9.3.3 Software Retirement Plan

The Retirement Plan documents all activities and decisions made on the retirement of the software. This document includes methods for accessing/retrieving of data, retention requirements, and media re-fresh schedules and procedure.

2.9.3.4 Data Change Request (DCR)

The DCR documents a formal request for a data modification that cannot be accomplished via the standard application user interface. Typically, changes that require a DCR are accomplished via macros, queries, or related methods. These changes often involve mass modifications to a set of data, but may be as simple as adding additional database entries to an existing table.

If no automated audit trail is in use for a data change, and there is no defined and approved alternate data update method in the Configuration Management Plan, and the change is one of the following types, then a DCR must be prepared for all software sources:

- classified data is modified;
- a data change which is a level 2 or 3 impact level (e.g., change of parameters on a life safety system); or
- a data change which has possible waste, fraud, or abuse implications (e.g., financial data, personnel records, security records) as defined by the system owner.

If an automated audit trail is in use for a change of the above type, then a DCR is optional. In all other cases, a DCR is fully optional.

The DCR must be approved at time of issuance not via the SEPRF.

2.9.3.5 Project Team Exemption Requests

For exemption requests that require disposition from the Project Team, the exemption rationale shall be documented. The documentation may be in the SPP, by an Exemption Request Form or in the comments field for the deliverable in SAM.

2.9.3.6 SRB Exemption Requests

For exemption requests that require disposition from the SRB, an Exemption Request Form or SAM Exemption Request must be submitted to the SRB. Exemption requests submitted to the SRB shall contain a statement of the deliverable that is being exempted and the rationale for the exemption. SRB exemption approvals/rejections shall be returned to the requester. The SRB minutes shall contain documentation of the exemption action.

2.9.3.7 Deliverable Approvals

Approvals of deliverables are based on the documents being reviewed and comments being resolved with responsible personnel prior to approval. Approval responsibility is assigned to the roles identified in Section 2.9.4. Approvals of the deliverables in this stage may be done via a signature on the document or the SEPRF at the discretion of the Project Team.

2.9.4 Roles and Responsibilities in the Retirement Stage

The System Owner has overall responsibility for the successful execution of the Retirement Stage. Listed below are the Project Team roles and the tasks that are normally the responsibility of those roles. The Project Manager may reassign tasks based upon a team member's knowledge of the subject, writing skills, available time, etc. Individuals may perform more than one team member function. Additional team members and/or additional approval responsibilities may be added to the matrix at the discretion of the Project Team.

2.9.4.1 System Owner

Responsibilities in the Retirement Stage include:

- Determines the software is to be retired.
- Updates the Software Inventory in SAM.
- Determines retention requirements for data and software.
- Identifies the data owner if necessary.
- Approves the Data Change Request via an electronic approval on the DCR or a signature.
- Approves the Software Retirement Checklist and the Software Retirement Plan via a signature on the document or a Stage Exit Position Response Form.

2.9.4.2 Project Manager

Responsibilities in the Retirement Stage include:

- Participates in completion of the Software Retirement Checklist.
- Approves the Software Retirement Plan via a signature on the document or a Stage Exit Position Response Form.

2.9.4.3 Project Team

Responsibilities in the Retirement Stage include:

- Concurs with PT deliverable exemptions.
- Concurs with SB deliverable exemption requests prior to the Project Manager submitting the request to the SRB.

2.9.4.4 Computing and Telecommunications Security Organization (CTSO)

In the Retirement Stage, CTSO reviews project deliverables for security implications at the discretion of the Cyber Security Program Manager.

2.9.4.5 CTSO Representative

IF protected software, THEN the responsibilities in the Retirement Stage include:

- Participates in completion of the Software Retirement Checklist.
- Approves the Software Retirement Plan via a signature on the document or a Stage Exit Position Response Form.

2.9.4.6 Software Review Board (SRB)

Responsibilities in the Retirement Stage include:

- Approves/rejects exemption requests for identified required deliverables.
- IF an Exemption Request form is used to request an exemption, THEN assigns an identification number to the exemption request.
- Reviews and approves or rejects requested exemptions, with rejection rationale.
- IF the Exemption Request form is used, THEN the exemption approval/rejection is marked, a copy is placed in the SRB files and the exemption request is returned to the Project Manager.

2.9.4.7 Lead Analyst

Responsibilities in the Retirement Stage include:

- Prepares the Software Retirement Checklist and the Software Retirement Plan.
- Creates and verifies the Software Backup Record Copies.
- Prepares and approves the Data Change Request.
- Approves the Software Retirement Checklist and the Software Retirement Plan via a signature on the document or a Stage Exit Position Response Form.

2.9.4.8 Quality Engineer (QE)

Responsibilities in the Retirement Stage include:

- Participates in completion of the Software Retirement Checklist.
- Approves the Software Retirement Plan via a signature on the document or a Stage Exit Position Response Form.

2.9.4.9 Product Engineer (PE)

IF product software, THEN the responsibilities in the Retirement Stage include:

- Participates in completion of the Software Retirement Checklist.
- Approves the Software Retirement Plan via a signature on the document or a Stage Exit Position Response Form.

2.9.4.10 Information System Security Officer (ISSO)

IF the software is on a classified system, THEN the responsibilities in the Retirement Stage include:

- Participates in completion of the Software Retirement Checklist.
- Approves the sanitization of electronic media that is no longer required in accordance with the AIS Security Handbook (Y19-401INS).
- Approves the Software Retirement Plan via a signature on the document or a Stage Exit Position Response Form when security controls for information are defined.

2.9.4.11 Records Analyst

Responsibilities in the Retirement Stage include:

- Participates in completion of the Software Retirement Checklist.
- Assists in the retention and disposition determinations and other Records Management functions.
- Approves the Software Retirement Plan via a signature on the document or a Stage Exit Position Response Form.

3 RECORDS

Records generated as a result of this procedure are maintained in accordance with BWXT Y-12 records management practices and established retention and disposition schedules. Records generated are owned by the Software Owner. The records include but are not limited to the following:

- Protected Software Identification (PSI) form
- Exemption Request
- Data Change Request
- Software Change Request
- Version Description Document
- Stage Exit Position Response Form
- Mission Essential Evaluation/Continuity of Operations Checklist
- Software Retirement Checklist
- Feasibility Study Document
- Risk Assessment
- Software Requirements Specification
- Software Project Plan
- Completed Units and Modules Code
- Integration and System Test Plan
- Application Security Specifications and Controls
- Continuity of Operations Test Plan
- Application Certification Test Plan
- Acceptance Test Plan
- Programmer's Reference Manual
- Code Review Documentation
- Configuration Management Plan
- Software Retirement Plan

4 Source Documents

- BWXT Y-12 Standards/Requirements Identification Document (S/RID), Requirement Units
- BWXT Y-12 Identification Documents (BWXT Y12ID) 9940
- Y60-WP-001, Weapons Quality Program Description
- QC-1, DOE-AL, Quality Criteria
- DOE Order 471.2A, Information Security Program
- DOE-STD-4001-2000, Design Criteria Standard for Electronic Records Management Software Applications
- DOE O 200.1-1, Information Management Program (36CFR, 1234)

5 References

- Y19-401INS, AIS Security Handbook
- DOE-STD-4001-2000, Design Criteria Standard for Electronic Records Management Software Applications
- Y60-802, Calibration and Control of Measuring and Test Equipment (M&TE)

Appendix A – Glossary

A

ACCEPTANCE TESTING

The activity performed to verify that the installed software meets the initial requirements for operation established by the System Owner.

ANALYST

An individual who is skilled and trained to define problems and to analyze, develop, and express algorithms for their solution, especially algorithms that may be resolved and implemented by a computer.

APPLICATION SOFTWARE

Software specifically produced for the functional use of a computer system; for example, software for navigation, gun fire control, payroll, general ledger.

AUTHORIZED DERIVATIVE CLASSIFIER (ADC)

Individual authorized to determine that documents and/or material are (a) unclassified or (b) classified as Restricted Data, Formerly Restricted Data, or National Security Information in accordance with existing guidance or source documents.

B

BASELINE

A set of configuration items (software components and documents) that has been formally reviewed and agreed upon, that serves as the basis for further development, and that can be changed only through formal change control process.

BUSINESS RULES

Statements that detail the activities in a business system and identify data required for these activities. These statements may be augmented with diagrams and matrices to assist in a clear understanding of the ways in which the activities and data of the business system interrelate. The business rules are designed to be understood by management, end users, developers, and analysts and to greatly increase communication among these groups.

Appendix A – Glossary

C

CLASSIFIED APPLICATION

An application that processes classified data or contains classified information as part of the application.

CLASSIFIED INFORMATION/DATA

Plain text or machine-encoded information/data that requires safeguarding in the interest of national security as determined by an Authorized Derivative Classifier.

COMMERCIAL OFF-THE-SHELF SOFTWARE (COTS)

Software, which is available to the general public for procurement and utilization as needed. This software is purchased and used as received as is within the Y-12 Complex.

CTSO REPRESENTATIVE

A person assigned by the Computing and Telecommunications Security Organization (CTSO) to represent the cyber security organization throughout the life cycle(s) of a protected software application.

This person recommends the certification of a protected software application for production status to the certifying authority. The certifying authority is the System Owner for applications residing on unclassified processors or the Information System Security Officer (ISSO) for applications residing on classified processors. The CTSO Representative will be the Lead Analyst of the application unless otherwise specified by the CTSO in writing.

CONFIGURATION CONTROL

An element of configuration management consisting of the evaluation, coordination, approval/disapproval, and implementation of changes to configuration items after formal establishment of their configuration identification.

CONFIGURATION MANAGEMENT

A discipline that effectively controls and manages all modifications to a software component, product, or system. Technical and administrative processes and tools are used to identify and document the functional and physical characteristics of the configuration items, manage and track changes to those items, record and report change processing and implementation status, and verify compliance with specified requirements. This is a Software Engineering Institute Capability Maturity Model key process area designed to establish and maintain the integrity of the software work products throughout the project's lifecycle.

Appendix A – Glossary

D

DESIGN AGENCY/GOVERNMENT AGENCY (DA/GA)

Software developed by another NNSA site or Government agency available for use at Y-12.

DEVELOPERS

Developers are programmers/analysts who design, implement, and modify an application software system.

DEVELOPMENT AREA

Computer area in which software is stored during development which excludes the possibility of production use. The development area is isolated from production use by system, physical, or administrative controls.

E

EMERGENCY CHANGE

A change to production software which abbreviates the normal change control process in order to correct an operational problem that disables a critical business activity.

EXEMPTION

The approval not to provide a deliverable specified in the deliverables matrix. Exemptions must have a justification and rationale. Exemptions may be granted by the SRB and the Project Team.

F

FUNCTIONAL SECURITY REQUIREMENTS

Those protection mechanisms required for application software to mitigate the exposures of delay, damage, disclosure, fraud, and theft as well as those that are required to be included based on the type of data or application.

I

INFORMATION

Refers to all data regardless of its physical form (i.e., paper printouts, tapes, disks or disk packs, memory chips, microfiche or microfilm, in communication lines, and on display terminals).

INFORMATION SYSTEM SECURITY OFFICER (ISSO)

Individual assigned the responsibility for ensuring the implementation of and compliance with all policies, standards, and procedures for the AIS Security Program for a single classified or Unclassified National Security Related (UNSR) processor or group of classified or UNSR processors. This individual certifies protected software projects that reside on a classified processor.

Appendix A – Glossary

L

LEAD ANALYST (LA)

Individual responsible for the analysis, design, planning and development of the software plus the technical leadership of the Project Team.

LOGICAL MODEL

A normalized model of data which represents the structural organization of data elements into logical records and the associations between those logical records. The logical model is independent of the individual applications of the data and of the software or hardware mechanisms that are used to represent and access the data. (Paraphrased from *Computer Data-Base Organization* by James Martin and *Applied Data Modeling in the IRM Environment* by William G. Smith).

LOGICAL RECORD

A record independent of its physical environment. Portions of the same logical record may be located in different physical records, or several logical records or parts of logical records may be located in one physical record.

M

MEASURING AND TEST EQUIPMENT (M&TE)

All of the measuring instruments, measurement standards, reference materials, and auxiliary apparatus that are necessary to perform a measurement. This term includes the measuring equipment used in the course of testing and inspection, as well as that used in calibration.

MIGRATION AREA

Area in which software is integrated with other software systems for testing and acceptance by the user representatives. The Migration Area is a simulated production environment and is isolated from production use by system, physical, or administrative controls.

MISSION-ESSENTIAL

An application that must remain operational for the company to perform its mission. If the normal processing computer or site suffers an outage, this application would need to be continued on a backup computer at the same or at an alternate site. An application is a candidate for being declared mission-essential if their delay in being processed may cause the company to incur a legal obligation or a financial penalty or to miss federal reporting requirements. This determination is made using the Protected Software Identification form.

MISSION-ESSENTIAL DATA

Any computerized information that, for the company to perform its mission, must be restorable if destroyed, is categorized as mission-essential. Mission-essential information can be classified or unclassified, or sensitive or non-sensitive. Essentially, the magnitude of the loss determines whether information is mission-essential.

Appendix A – Glossary

P

PRODUCT ENGINEER (PE)

An engineer who has oversight responsibility for the weapons program with which the software is associated.

PRODUCT SOFTWARE

Any software utilized during the research, design, development, procurement, production, maintenance, stockpile evaluation, dismantlement, and/or disassembly/disposal of weapons and weapons-related material. Product software may be utilized for manufacturing control, processes, inspection, and test of material; record keeping operations that determine or document the quality and conformance to the design intent of the shipped product; to control calibration of standards and measurement devices; to provide analysis capability to determine product acceptability; to control the transfer or maintenance of design definition; and to control the function of weapon and weapon-related components. The determination of product software is done by mutual agreement between the Product Engineer, Quality Engineer, and the operating division and will follow the guidelines for determining product procedures.

PRODUCTION AREA

Area from which software is invoked by the user for production purposes.

PRODUCTION USE

The point in the life cycle (Maintenance) where formal configuration control of the product baseline begins and software is released to users for performance of its intended function.

PROGRAMMER

The individual who performs the implementation activities for the software or system design.

PROJECT FILE

A central repository(ies) of material pertinent to a project. Contents typically include all work products, memos, plans, technical reports, and related items.

PROJECT MANAGER (PM)

The individual with total business responsibility for all software activities of a project. The project manager directs, controls, administers, and regulates a project.

PROJECT TEAM (PT)

The project manager, analysts, programmers, and other staff assigned as the core group for a project. The Project Team may include representatives of the other functional areas (e.g., technical writer and telecommunications expert) responsible for contributing to the development, installation, and maintenance of the software product.

Appendix A – Glossary

PROTECTED SOFTWARE

Application software that is sensitive or classified because of the risk and magnitude of the loss or the harm that could result from improper operation, deliberate manipulation, or interruption of service; processes sensitive or classified data; or resides on a classified processor *and* meets thresholds that describe its importance or impact on Y-12. Refer to the Protected Software Identification Form for full definition.

Q

QUALITY ENGINEER (QE)

Responsible for independent SQA oversight for software.

QUALITY EVIDENCE DATA

Information that indicates the extent of conformance to requirements and control over processes. This information is based on process control data, inspection and acceptance activities, or traceability data.

R

RECERTIFICATION

An ongoing reassurance that a previously certified application has been periodically reviewed, that compliance with established protection policies and procedures remains in effect, and that security risks remain at an acceptable level.

RECOVERY

The restoration of a computer or communications system to normal operations following an interruption of service. This process may make use of alternative equipment or different facilities to resume operation on a temporary basis until the original system can be brought back to normal operating status.

REQUIREMENTS TRACEABILITY MATRIX

A table used to ensure that activities and work products can be traced to the project requirements (from the point of identification through implementation) as described in the Software Requirements Specification. A copy of the matrix is maintained in the project file. The matrix is created during the requirement stage and expanded in following stages.

Appendix A – Glossary

S

SAFETY ANALYSIS & DESIGN SOFTWARE

Computer software that is not part of a safety SSC (structures, systems, and components), but is used in the safety classification, design, and analysis of nuclear facilities to:

Ensure the proper accident analysis of nuclear facilities;

Ensure the proper analysis and design of safety SSCs;

Ensure the proper identification, maintenance, and operation of safety SSCs.

SAFETY SYSTEM SOFTWARE

Computer software and firmware that performs a safety system function as part of a SSC that has been functionally classified as Safety Class (SC) or Safety Significant (SS). This also includes computer software such as human-machine interface software, network interface software, PLC programming language software, and safety management databases that are not part of an SSC but whose operation or malfunction can directly affect SS and SC SSC function.

SECURITY SPECIFICATIONS

A detailed description of the safeguards required to protect a system.

SENSITIVE APPLICATION

An application that processes sensitive information as part of the software or is sensitive because of the risk and magnitude of the loss or the harm that could result from improper operation, deliberate manipulation, or interruption of service.

SENSITIVE INFORMATION

Information that a competent authority has determined must be protected because its unauthorized disclosure, alteration, loss, or destruction will at least cause perceivable damage to someone or something.

SOFTWARE

Computer programs, procedures, and associated documentation and data pertaining to the operation of a software product or system. Software includes applications, operating systems, NC programs, software routines and macros, firmware, Programmable Logic Controller (PLC) Ladder Diagrams; M&TE associated software and exploratory projects.

SOFTWARE ACQUISITION

The process of identifying and obtaining software for use in Y-12 that meets this Y80-102 instruction.

SOFTWARE APPLICATION ACRONYM

A unique acronym to identify a specific software application. A maximum of ten-character alphanumeric string (including "-"and "/") with no embedded spaces.

Appendix A – Glossary

SOFTWARE APPLICATION MANAGER (SAM)

The Web-based software established to support implementation of the requirements in this Instruction. Web based software utilized by roles identified in this document to manage a software project and track compliance.

SOFTWARE COMPONENT

Any source module, executable or load module, file definition, data base definition, screen definition, command language procedure, or build procedure for a software application.

SOFTWARE COMPONENTS LIST

A list of all software components for a software application that establishes the unique identity of each component using a date-time stamp or version identifier.

SOFTWARE DOCUMENTATION

Documentation that includes the overall structure of the system, the source code listings, the definitions of physical data structures, and the instructions necessary to compile, link, and execute all programs. The descriptions of unusual conditions, constraints, interfaces to other systems, and error handling should also be included.

SOFTWARE FAILURE IMPACT (SFI) LEVEL

A numerical categorization of software based on identified consequences of failure that determines the applicable requirements and configuration management within each of the life-cycle stages. Software must be assigned to the highest failure impact for the threshold met.

SOFTWARE OWNER

BWXT Y-12 employee responsible for the software throughout the software life cycle. The Software Owner is responsible for working with the funding source to assure adequate funding for the software implementation and on-going maintenance.

SOFTWARE PROJECT

An undertaking involving any of the following components: research, development, acquisition, modification, or implementation of a software system.

SOFTWARE QUALITY ASSURANCE (SQA)

The planned and systematic actions necessary to provide adequate confidence that software will perform satisfactorily in service.

SOFTWARE REVIEW BOARD (SRB)

The body responsible for supporting the SQA Manager in oversight of the Software Management Program in Y-12. This board performs the necessary and appropriate reviews and approvals. The SRB is responsible for reviewing and approving failure impact levels and exemptions of some deliverables. The SRB membership is provided in the SRB Charter, Appendix B.

Appendix A – Glossary

SOFTWARE SOURCE

The source of software implemented to address a specific set of requirements. Examples: COTS, design agency or government agency (DA/GA), developed internally, developed externally.

SOFTWARE TYPE

The basic function performed by the application when executed. Examples: word processing, data entry, database, spreadsheet, process control, data acquisition, analysis, software generation, operating system.

SOFTWARE USE

The business functions performed by the software when implemented in Y-12. Examples: administrative, facility, product, Work for Others.

SQA MANAGER

The SQA Manager by oversight of the Software Management Program interprets the requirements of Y80-101PD and Y80-102, assists personnel in the implementation of the requirements defined in the Y80-102, evaluates the effectiveness of activities, coordinates the software audit program, represents Y-12 in matters involving external organizations, and supervises the SQA training and awareness programs. The SQA Manager reports functionally in the Quality Assurance Division.

SYSTEM

(1) A collection of people, machines, and methods organized to accomplish a set of specific functions.

(2) An integrated whole that is composed of diverse, interacting, specialized structures and sub functions.

(3) A group or subsystem united by some interaction or interdependence, performing many duties but functioning as a single unit.

SYSTEM ENVIRONMENT CHANGE

Any change to hardware, system or network software, or any other component on which a system depends on to execute.

U

USER POINT OF CONTACT

An individual who represents the users throughout the application life cycle on the Project Team.

USER

An organization or individual who utilizes software or computer systems.

Appendix A – Glossary

V

VERSION IDENTIFICATION

A standard version numbering method is needed for all software at Y-12 to ensure consistency in software identification. The suggested version identification method for a specific release of a software application that is developed internally or by external sources is as follows:

AAAAAAAA Vxx.yy.zzz (e.g., SMCS v01.00.001) where

AAAAAAAA is the product acronym,

xx is the major version number,

yy is the minor version number,

zzz is the release number.

Major version number is set to “1” upon initial release of the software product version for production use. Thereafter, it is incremented whenever an enhancement to the software is to be performed which represents a major functionality change.

Minor version number is set to “0” upon initial release or whenever there is an incremental change of the major version number. Thereafter, the minor version number is incremented whenever an enhancement to the software is to be performed which represents a minor functionality change.

Release number is set to “1” upon initial release. It is incremented whenever the configuration of the production software is changed in any way.

Minor changes which do not affect the functionality of the software may be signified by only incrementing the release number. Any change to software functionality must be signified by the incrementing of the major version number or the minor version number.

The identification scheme of purchased COTS or software from Design Agencies or Other Government Agencies may be used when it conflicts with the identification scheme defined in this instruction.

Appendix A – Glossary

W

WEAPON MATERIAL

DOE nuclear weapons, assemblies, components, software or parts thereof.

WEAPON-RELATED MATERIAL

Any material, including associated software and test and handling equipment, other than weapons material being developed and produced for or by the DOE and intended for use in conjunction with, or in any way related to, weapons.

WORK FOR OTHERS SOFTWARE

Software which is developed for use by non-DOE sponsors but which is contracted in an Inter-agency Agreement with DOE.

Appendix B - Y-12 Software Review Board (SRB) Charter July 1, 2004

ARTICLE I – NAME

The name of this organization shall be the Y-12 SRB, hereafter referred to as the SRB.

ARTICLE II – OBJECTIVES

The primary objective of the SRB is to support the implementation of Y80-102 relating to Software Failure Impact determinations and disposition deliverable exemption requests. To accomplish the primary objective in a manner consistent with the requirements of the Y80-102, the SRB:

- Conducts routine meetings to review selected project documentation; to review determination of Software Failure Impact; and to review for approval applications submitted for exemption from specific deliverables. The project documentation may be selected by the SQA Manager or submitted by the Project Manager. The documentation review is to verify correctness and accuracy consistent with the established requirements.

- Provides for the documentation of decisions made by the SRB, and for the dissemination of those decisions to appropriate parties.

ARTICLE III – MEMBERSHIP

The SRB is composed of Subject Matter Experts from Cyber Security, Quality (SQA Manager, Product QE and Facility QE), Product Engineering and various organizations at Y-12.

The SRB may at their discretion utilize input from the following individuals and/or organizations: Requester, Criticality Safety, Safety, Records Management, Auditing, RADCON, Quality Engineering, and Facility Engineering.

The Y-12 Quality Manager is responsible for ensuring SRB participation. SRB members are nominated by their respective Division Manager.

Each member of the SRB shall be entitled to one vote on any matter of business to be so decided. The vote may be cast by the SRB member, or by a proxy designated by the SRB member.

Appendix B - Y-12 Software Review Board (SRB) Charter July 1, 2004

ARTICLE IV – SRB OFFICERS

The officers of the SRB shall be a Chair, a Vice-Chair and a Board Secretary. Each officer of the SRB shall be a member of the board.

The Chair for the SRB is to be named by the SQA Manager.

The Vice-Chair office shall be held by an SRB member as elected by the SRB.

The Board Secretary office shall be held by an SRB member as appointed by the SQA Manager.

The SRB Chair shall:

- be the principal executive officer of the SRB;
- designate SRB subcommittees and subcommittee chairs;
- call for and preside over all SRB meetings;
- assume responsibility for SRB communications and, particularly, for recommendations and publications of the SRB;
- Appoint a committee member to serve as acting Board Secretary in the event that the Board Secretary is not in attendance.

The Vice-Chair shall act as Chair pro tem in the absence of the SRB Chair.

The Board Secretary shall:

- be responsible for all correspondence addressed to or originating from the SRB, except as otherwise provided, and shall forward a copy of all correspondence to each board member;
- keep the minutes of meetings for long-term retention and publish same after each meeting (including such attachments and supplementary material presented at the meetings as may be deemed appropriate);
- maintain an historical record of all SRB documents (Bylaws, Internal Procedures, etc.) and all decisions made by the SRB;
- maintain a list of SRB members, their affiliate organizations, and their mail name.

The SQA Manager shall act as Chair pro tem in the absence of the SRB Chair and Vice-Chair. The SQA Manager will maintain the list of approved SRB membership.

Appendix B - Y-12 Software Review Board (SRB) Charter July 1, 2004

ARTICLE V – MEETINGS

Prior to conducting any formal SRB business, the SRB Chair, SRB Vice-Chair or the SQA Manager must establish that a quorum exists. A quorum shall be declared only if the following conditions are satisfied:

- The SRB Chair, Vice-Chair or SQA Manager is in attendance.
- The PE or Product QE representative (or proxy) is in attendance.
- The CTSO representative (or proxy) is in attendance.
- At least half the other SRB members (or designated proxy) are in attendance.

Each SRB member shall have the right to vote at all meetings on all matters pertaining to the general affairs of the SRB and on which a vote of the members is required.

At all duly constituted meetings of the SRB all questions, except as otherwise provided in these bylaws, shall be decided by the vote of a majority of the SRB members present at the meeting.

When deemed appropriate by the Chair, or upon a motion duly made and seconded by a SRB member, a straw vote may be conducted to obtain a consensus on any business before the SRB. The vote may be cast by hand, and each person in attendance shall be entitled to one vote. The result of the straw vote to obtain a consensus of those present is not binding upon the SRB members in their final vote on the question.

The procedures and rules contained in Robert's Rules of Order shall govern the conduct of SRB meetings in all cases to which they are applicable, and in which they are not inconsistent with this Charter and the Internal SRB Procedures document.

ARTICLE VI – SUBCOMMITTEES

The Chair shall designate and dissolve SRB subcommittees as directed by the SRB membership.

The Chair of each subcommittee shall be appointed by the SRB Chair.

Each subcommittee chair shall report any actions of the committee to the membership at the next meeting.

ARTICLE VII – AMENDMENTS TO THE BYLAWS

The SRB may create Bylaws governing SRB business practices. Amendments to these Bylaws may be proposed to the Board Secretary by any SRB member. The text of the proposed amendment to the Bylaws shall be made available to all SRB members at least three working days prior to its discussion at a SRB meeting.

Amendment of these Bylaws requires an affirmative vote of two thirds of the votes cast by attending SRB members and approval by the SQA Manager.